

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR
INGENIERÍA EN INFORMÁTICA

Series Notification System

Definición de un sistema de notificaciones personalizable por el usuario

Autor: César de la Cruz Rueda
Tutor: Sara Tena García
Co-tutor: David Díez Cebollero

Agradecimientos

Este proyecto y todo lo que ha llevado a la realización del mismo no habría sido posible sin el apoyo, ayuda y paciencia de muchas personas y es por ello que quiero agradecerse a todos ellos, aunque seguro que me dejaré a alguno.

En primer lugar a Sara y David, mis tutores del proyecto por haber confiado en mi idea cuando ya casi estaba decidido a que no podría realizarse, por darme tan buenos consejos y ser tan pacientes a la hora de realizar el seguimiento y revisión del proyecto.

A todos los compañeros con los que he compartido prácticas, clases, horas y horas en cafeterías, pasillos y otros lugares para poder terminar a tiempo las interminables prácticas de la carrera, mención especial a José Miguel, Erick, José y David con lo que he compartido muchos y muy buenos ratos.

A mis abuelos y mi tía, que han sido un apoyo increíble durante todos los años que he pasado en la universidad, y sin cuyo cariño seguro que no hubiera encontrado la fuerza para seguir adelante en ciertos momentos.

A mis padres que han sido las personas que me han hecho como soy, que me han educado y dado la posibilidad de estar aquí y sin cuyos ánimos, apoyo e insistencia no habría tenido la entereza de haber seguido adelante en ciertos momentos.

Y por supuesto a Bea que desde hace más de 8 años ha sido mi apoyo y que ha sufrido más que nadie las interminables horas de prácticas y que desde el principio me ha dado el amor y la fuerza para seguir adelante, que me ha hecho ser mejor persona y ha sabido animarme cuando las cosas no salían bien y darme apoyo durante todo el camino.

A todos, muchísimas gracias.

Resumen

El presente proyecto se engloba dentro del contexto de los sistemas de notificaciones, se puede definir una notificación como la acción y efecto de dar noticia de algo con un cierto propósito. Un sistema de notificaciones se encarga de notificar de forma automática a los usuarios cuando un determinado evento ocurre.

El auge de los dispositivos móviles supone una gran oportunidad en el mundo de los sistemas de notificaciones puesto que permite que los usuarios puedan ser notificados en tiempo real y en cualquier lugar.

Con este proyecto se pretende buscar solución a uno de los mayores problemas existentes en los sistemas de notificaciones: la generación de notificaciones sin control o configuración del usuario que provoca el hecho de que se lleven a cabo muchas notificaciones que no sean de interés para el usuario. Para ello, se pretende dotar al usuario con un papel más participativo e interactivo que le permita configurar las notificaciones a recibir. De este modo, a su vez, se pretende conseguir mejorar la experiencia de usuario en los sistemas de notificaciones.

A fin de alcanzar el propósito del proyecto, y debido al amplio uso de los sistemas de notificaciones en distintos dominios, se ha realizado una prueba de concepto de una aplicación móvil que permite la configuración de notificaciones a los usuarios sobre la emisión de episodios de series. A través de la aplicación, los usuarios podrán buscar y seleccionar las series sobre las que quieren recibir notificaciones, así como configurar cuándo recibirlas. Adicionalmente, se ofrecerá la posibilidad al usuario de interactuar con el sistema permitiendo enviar *feedback* sobre los episodios vistos.

Abstract

This project is framed in the context of notification systems. A notification can be defined as “the action and effect of giving notice of something with a purpose”. Notifications systems ensure automatically notify users when a particular event occurs.

The rise of mobile devices implies a great opportunity in the world of notification system as it allows the user to be notified in real time and anywhere. This project aims to find a solution to one of the biggest problems in systems notifications: the notification generation either uncontrolled or without user settings causing a lot of notifications that are not of interest to the user. With such purpose, this project aims to give the user a more relevant and interactive role, allowing the user to configure which notification wants to receive. In this way, it is aimed additionally improving the overall the user experience in notification systems.

Due to the wide use of notification systems in different domains and with the aim of achieving the main goal of the project, it has been developed a proof of concept of a mobile application that allows users to configure notifications about the broadcast of series episodes. By the using the application, users could search and select the series on which they want to receive notifications and to configure when to receive them. Additionally, the application would offer the possibility to interact with the system for sending feedback about watched episodes.

Índice de contenidos

Agradecimientos	2
Resumen	3
Abstract	4
Índice de contenidos	6
Índice de figuras	7
Índice de tablas	8
Glosario de términos	9
1 Introducción	10
1.1 Definición del problema	12
1.2 Objetivos	13
1.3 Fases del desarrollo	14
1.4 Estructura de la memoria	15
2 El estado de la cuestión	16
2.1 Sistemas Operativos Móviles	16
2.2 Aplicaciones de gestión y notificación de eventos	26
3 Gestión de proyecto software	30
3.1 Plan de trabajo	31
3.2 Gestión de recursos	33
3.3 Presupuesto	36
4 Solución	39
4.1 Descripción de la solución	39
4.2 El proceso de desarrollo	40
5 Evaluación	83
5.1 Plan de Pruebas	83
6 Conclusiones	89
6.1 Aportaciones realizadas	89
6.2 Trabajos futuros	89
6.3 Problemas encontrados	89
6.4 Opiniones personales	90
7 Bibliografía	91
Anexo I. Control de versiones	94
Anexo II. Seguimiento de proyecto fin de carrera	95
Anexo V. Cuestionarios de evaluación	99

Índice de figuras

Ilustración 1: gráfica evolutiva uso SO Móviles.....	18
Ilustración 2: gráfica uso total SO móviles.....	18
Ilustración 3: Tabla de ventas dispositivos móviles por S.O.....	20
Ilustración 4: gráfico número de aplicaciones	21
Ilustración 5: esquema general de la aplicación	50
Ilustración 6: Esquema de la base de datos web	51
Ilustración 7: Esquema de la base de datos del móvil.....	52
Ilustración 8: Diagrama de clases de la parte web.....	60
Ilustración 9: Diagrama de clases de la parte móvil.....	61
Ilustración 10: Diagrama de clases móvil: vistas	62
Ilustración 11: Diagrama de clases móvil: model y service	64
Ilustración 12: Diagrama de clases móvil: utils.....	65
Ilustración 13: Prototipo de interacción, búsqueda.....	66
Ilustración 14: Prototipo de interacción, series favoritas	67
Ilustración 15: Pantalla principal	68
Ilustración 16: resultados búsqueda	68
Ilustración 17: ficha serie.....	68
Ilustración 18: Series Favoritas.....	69
Ilustración 19: Nuevos episodios	69
Ilustración 20: Pantalla principal	70
Ilustración 21: resultados de búsqueda	70
Ilustración 22: pantalla de progreso.....	70
Ilustración 23: ficha de serie.....	70
Ilustración 24: series favoritas	71
Ilustración 25: cuadro de confirmación.....	71
Ilustración 26: nuevos episodios.....	72
Ilustración 27: Pantalla principal, búsqueda.....	73
Ilustración 28: resultado de búsqueda	73
Ilustración 29: ficha de serie.....	74
Ilustración 30: mensaje de confirmación	74
Ilustración 31: series favoritas	75
Ilustración 32: top series	75
Ilustración 33: preferencias	76
Ilustración 34: dialogo preferencia, frecuencia.....	76
Ilustración 35: estructura del proyecto web.....	79
Ilustración 36: estructura de la aplicación android	80
Ilustración 37: estructura de la aplicación android: directorio src.....	81
Ilustración 38: estructura de la aplicación android: directorio res	82

Índice de tablas

Tabla 1: Tabla comparativa aplicaciones por S.O.	21
Tabla 2: comparativa características/S.O.	24
Tabla3: comparativacaracterísticas/aplicación	29
Tabla 4: estimación de las tareas	32
Tabla 5: asignación de recursos a tareas	35
Tabla 6: dedicación en días.....	36
Tabla 7: coste personal.....	36
Tabla 8: coste de recursos materiales.....	37
Tabla 9: coste global del proyecto.....	38
Tabla 10: coste post beneficio y riesgo.....	38
Tabla 11: Coste final del proyecto.....	38
Tabla 12: Requisitos Funcionales.....	41
Tabla 13: Requisitos no funcionales.....	42
Tabla 14: tabla de casos de uso.....	43
Tabla 15: Caso de uso CU-001.....	44
Tabla 16: Caso de uso CU-002.....	44
Tabla 17: Caso de uso CU-003.....	45
Tabla 18: Caso de uso CU-004.....	45
Tabla 19: Caso de uso CU-005.....	45
Tabla 20: Caso de uso CU-006.....	46
Tabla 21: Caso de uso CU-007.....	46
Tabla 22: Caso de uso CU-008.....	46
Tabla 23: Caso de uso CU-009.....	47
Tabla 24: Caso de uso CU-010.....	47
Tabla 25: Caso de uso CU-011.....	48
Tabla 26: Caso de uso CU-012.....	48
Tabla 27: Caso de uso CU-013.....	48
Tabla 28: Caso de uso CU-014.....	49
Tabla 29: Caso de uso CU-015.....	49
Tabla 30: lista de casos de pruebas.....	85
Tabla 31: resultados de las pruebas.....	86
Tabla 32: resumen participantes.....	88
Tabla 33: resumen resultados.....	88

Glosario de términos

Catálogo de términos específicos del contexto del trabajo.

- **API:** Interfaz de programación de aplicación (Application Programming Interface en sus siglas en inglés) se trata de un conjunto de funciones o procedimientos que ofrece una aplicación o librería para poder ser utilizada por otro Software.
- **APK:** Formato de archivo que se usa para empaquetar aplicaciones para el sistema operativo para móviles Android.
- **Bluetooth:** especificación para redes inalámbricas que permite la transmisión de voz y datos entre dispositivos mediante radiofrecuencia.
- **GPS:** Sistema de posicionamiento Global (Global Positioning System en inglés) es un sistema de navegación global por satélite que permite determinar la posición de un objeto en todo el mundo.
- **HTTP:** Protocolo de transferencia de Hipertexto (Hiper Text Transfer Protocol en Inglés) es el protocolo usado en las comunicaciones en Internet.
- **IDE:** Entorno de Desarrollo Integrado (Integrated Development Environment en Inglés) es un programa informático compuesto por un conjunto de herramientas que permiten realizar diferentes tareas.
- **LED:** Diodo emisor de luz (Light Emitting Diode en Inglés) es un diodo semiconductor que emite luz.
- **NDK:** Equipo de Desarrollo Nativo (Native Development Kit en Inglés) es un conjunto de herramientas que permite desarrollar directamente en una plataforma en vez de hacerlo sobre una máquina virtual.
- **SDK:** Equipo de desarrollo (Software Development Kit en Inglés) es un conjunto de herramientas que permiten desarrollar aplicaciones en un sistema concreto.
- **Smartphone:** Teléfono móvil que ofrece la mayor capacidad de conectividad y de computación y que permite instalar aplicaciones.
- **WiFi:** Es un mecanismo de conexión de dispositivos electrónicos de forma inalámbrica.

1 Introducción

El presente proyecto se engloba en el contexto de los sistemas de notificaciones. De forma general se puede definir una notificación como la acción y efecto de dar noticia de algo con un cierto propósito [1][2]. En el ámbito de la informática, un sistema de notificaciones es una combinación de software y hardware que proporciona un mecanismo para enviar un mensaje o información de valor, a un conjunto de destinatarios, en base a la ocurrencia de un evento específico [3][4]. En cambio, desde un punto de vista más orientado a la experiencia del usuario, un sistema de notificaciones puede ser definido como una interfaz típicamente usada en una situación de división de multitarea, cuyo objetivo es enviar información de valor y actual de diferentes modos y de forma efectiva y eficiente [5].

Independientemente de la perspectiva a aplicar, de manera general, los sistemas de notificaciones se componen, principalmente, de dos partes: los publicadores/productores, que son los encargados de crear o publicar el contenido y cuyo interés es que dicho contenido sea distribuido de forma rápida, a cuantas más personas mejor; y los subscriptores/consumidores, que buscan recibir información relevante de forma automática. Adicionalmente, de manera general los sistemas de información se caracterizan por ofrecer las siguientes funcionalidades:

- **Planificación:** el hecho de permitir decidir a los subscriptores cuando quieren recibir las notificaciones.
- **Interacción:** permitir feedback de los subscriptores.
- **Prioridad:** poder definir prioridades a ciertas notificaciones para ser tratadas de una manera diferente dependiendo del caso.
- **Escalado:** permitir que se varíe la prioridad de una notificación en tiempo real si la situación cambia o en base al feedback de otros subscriptores.
- **Tolerancia a fallos:** en ciertos casos se debe asegurar la disponibilidad del sistema mediante sistemas de redundancia para asegurar que el sistema opera de forma continua.

Los beneficios de un sistema de notificaciones más destacados incluyen la rápida disponibilidad de información relevante, el acceso casi inmediato a la comunicación [5] y el bajo coste de operación. Como consecuencia, los sistemas de notificaciones son comúnmente utilizados en muchos dominios distintos, tales como:

- **Medicina:** pudiendo ser aplicado a la notificación a pacientes, como por ejemplo ante la proximidad de una cita.
- **Seguridad:** pudiendo ser integrado en un sistema de alarma integral, notificando a los usuarios.
- **Servicios de emergencias:** permitiendo realizar notificaciones de posibles accidentes en las carreteras para evitar colapsos en las carreteras.
- **Banca:** permitiendo la notificación a los clientes de los movimientos en sus cuentas.
- **Comercio electrónico:** permitiendo notificar a los usuarios si ciertos productos elegidos alcanzan un precio.

Durante muchos años, los sistemas de notificaciones han sido utilizados en la web, permitiendo al usuario recibir notificaciones en su ordenador mientras realizaban otras tareas sobre contenidos seleccionados dentro de web y elegidos por el usuario. Ahora, con el auge de los dispositivos móviles inteligentes, se ha abierto una vía fácil de comunicación con los usuarios, que

asegura que las notificaciones serán recibidas se encuentre donde se encuentre el usuario, puesto que lo común es llevar estos dispositivos siempre.

Prueba de ello son aplicaciones como Gmail, Google Calendar o Facebook que inicialmente fueron desarrolladas como aplicaciones web, pero que han sido posteriormente desarrolladas para dispositivos móviles.

1.1 Definición del problema

En la mayoría de las ocasiones nos encontramos con que los sistemas de notificaciones generan abundantes notificaciones o notificaciones no deseadas, lo que puede distraer o molestar al usuario. Este hecho conlleva una alteración de la experiencia del usuario, distrayendo al usuario de su objetivo y no aportando valor real.

Se define como **experiencia de usuario** el conjunto de factores y elementos relativos a la interacción del usuario, con un entorno o dispositivo concretos, cuyo resultado es la generación de una percepción positiva o negativa de dicho servicio, producto o dispositivo [6]. Según Nielsen [8], la experiencia de usuario abarca todos los aspectos de la interacción del usuario final, siendo el primer requisito para una experiencia de usuario ejemplar conocer las necesidades del usuario sin molestarle ni agobiarle.

La experiencia de usuario es un concepto relativamente reciente y pretende unificar y ampliar los conceptos de usabilidad, diseño centrado en el usuario e interacción humano-maquina.

- La **usabilidad** se define como facilidad de uso de un sistema, así como de los métodos para mejorar la facilidad de uso de los sistemas [8]
- El **Diseño Centrado en el Usuario** es una filosofía de diseño que tiene por objetivo la creación de productos que resuelvan necesidades concretas de sus usuarios finales, consiguiendo la mayor satisfacción y mejor experiencia de uso posible con el mínimo esfuerzo de su parte [10].
- La **Interacción Humano-Máquina**, es la disciplina relacionada con el diseño, evaluación e implementación de sistemas informáticos interactivos para el uso de seres humanos, y con el estudio de los fenómenos más importantes con los que está relacionado [11]

Siguiendo la filosofía del diseño centrado en el usuario, los sistemas de notificaciones deberían tratar de conseguir la satisfacción y mejor experiencia de uso posible, sin que esto suponga un esfuerzo por parte del usuario. No obstante, aunque como se ha expuesto previamente, los sistemas de notificaciones se deben caracterizar por: permitir planificar las notificaciones, ser interactivos, permitir definir prioridad y ser escalables, la mayoría de los sistemas de notificaciones actuales no permiten a los usuarios planificar las notificaciones, sino que notifican al usuario tan pronto como se dispone de la información y no permite filtrar o priorizar la información que quieren recibir.

Es por ello que el principal problema que nos encontramos en los sistemas de notificaciones es que el usuario suele tomar un papel de sujeto pasivo frente a las notificaciones, recibiendo mucha información, pero sin poder elegir qué, cuándo, ni cómo. Sin embargo, en los últimos años, los usuarios de los sistemas informáticos han pasado de ser unos meros espectadores, a ser una parte relevante en los sistemas, requiriendo de su participación para crear el propio sistema o siendo directamente los creadores del propio contenido. Este cambio de rol de los usuarios, en los sistemas de notificaciones móviles, supone que los usuarios deberían ser capaces de configurar, desde su móvil, qué información quieren recibir, así como cuándo, cómo y con qué frecuencia.

1.2 Objetivos

El objetivo principal que se busca con la realización de este proyecto es permitir la gestión de notificaciones, a través de una aplicación móvil, ofreciendo un papel más decisorio al usuario. Este objetivo radica en el desarrollo una prueba de concepto de un sistema de gestión de notificaciones móvil sobre nuevos episodios de series.

Para la consecución del objetivo principal, los sub-objetivos básicos que el sistema deberá cumplir son:

- Permitir la búsqueda elementos.
- Permitir la selección de elementos al usuario para indicar sobre qué quiere ser notificado.
- Permitir seleccionar al usuario la frecuencia y hora en las que se quiere recibir las notificaciones.
- Realizar notificaciones sobre los elementos seleccionados a la hora y días seleccionados por el usuario.
- Permitir al usuario indicar al sistema que elementos ya han sido considerados, así como proporcionar feedback sobre estos elementos.

1.3 Fases del desarrollo

El proyecto se ha dividido en diferentes fases a la hora de desarrollarse, dichas fases componen el ciclo de vida de un proyecto de Software y recogen desde el inicio del proyecto hasta su finalización.

Las distintas fases son:

- **Definición de necesidades:** en esta fase se estudian los requerimientos y objetivos de la aplicación para saber qué es lo que la aplicación debe hacer.
- **Estudio de la situación actual:** en esta fase se estudian los sistemas y tecnologías actuales para conocer su estado, valorar las alternativas y seleccionar la tecnología con la que desarrollará el proyecto.
- **Análisis del sistema:** En esta fase se estudian y tratan de definir en detalle los requisitos del sistema, así como los elementos que intervienen en el sistema y las posibles limitaciones.
- **Diseño del sistema:** En esta fase se define como se va a realizar el proyecto una vez detallado el qué. Se define de forma detallada los elementos que compondrán el sistema, su arquitectura, el modelo de datos, la interfaz de acceso a los datos externos, así como la interfaz de usuario.
- **Implementación:** En esta fase es donde se pone en práctica el diseño de la aplicación realizado. Para ellos se utilizan las herramientas y tecnologías seleccionadas en las fases previas. La implementación debe cubrir los requisitos especificados en la fase de análisis.
- **Pruebas y Validación:** En esta fase se comprueba que el sistema cumple con los requisitos definidos y que, además, el resultado es satisfactorio para los usuarios potenciales.

1.4 Estructura de la memoria

La memoria realizada se compone de siete apartados:

- **Introducción:** presenta una breve introducción al proyecto, a los objetivos que se pretenden alcanzar en la finalización del mismo, la estructura de la memoria y la metodología utilizada durante la realización del proyecto.
- **Estado de la cuestión:** en este apartado se realiza el estudio de los sistemas de notificación y de las tecnologías más apropiadas para realizar dicho sistema.
- **Gestión del proyecto Software:** describe la gestión realizada del proyecto a desarrollar, incluyendo el alcance del proyecto, el plan de trabajo y la gestión de recursos.
- **Solución:** En este apartado se expone la solución elegida para contemplar los objetivos fundamentales del proyecto. En este apartado se describe a fondo el proceso de desarrollo que se seguirá a la hora de realizar el proyecto (análisis, diseño, pruebas, etc.).
- **Evaluación:** se presenta la evaluación del proyecto, donde se valida el plan de pruebas anteriormente presentado y se realiza una validación con usuarios.
- **Conclusiones:** se exponen las conclusiones que se han obtenido tras la realización del proyecto.
- **Bibliografía:** incluye toda la bibliografía usada para la realización del proyecto.

2 El estado de la cuestión

En este apartado se presenta un estudio de las tecnologías relacionadas con el proyecto desde dos ámbitos diferentes.

En primer lugar, se estudian los Sistemas Operativos para Smartphones que se pueden encontrar en el mercado, estudiando cantidad de usuarios, aplicaciones y documentación, así como sus características técnicas para seleccionar el Sistema Operativo móvil más adecuado.

En segundo lugar, con el objetivo de extraer las características esenciales para una aplicación de notificaciones, se estudian algunas de las aplicaciones que se encuentran disponibles relacionadas con los sistemas integrados de notificaciones y las características que disponen en relación con las notificaciones y su gestión.

2.1 Sistemas Operativos Móviles

En este apartado se describen las opciones de Sistemas Operativos (S.O.) que podrían seleccionarse para la implementación de una aplicación para dispositivos móviles inteligentes.

Se pretenden estudiar aquellos Sistemas Operativos móviles con mayor grado de madurez, uso y cuya tendencia de crecimiento en número de terminales vendidos y de aplicaciones creadas fuera ascendente en el último año. Además se ha considerado la documentación y comunidad de desarrolladores existente. Por ello, se ha recopilado información sobre estudios estadísticos de uso, de ventas, número de aplicaciones, así como de la documentación existente. De esta manera el lector podrá hacerse una idea de la evolución, crecimiento e importancia de cada uno de los S.O.

Uso

- Las estadísticas de uso indican con que SO se realizan visitas a los sitios webs o aplicaciones que disponen de un sistema de control de estadísticas. Se ha optado por obtener los datos de statcounter[1] <http://lema.rae.es/drae/?val=notificacion>, accedido el 22 de Diciembre de 2012
- [2] <http://lema.rae.es/drae/?val=notificar>, accedido el 22 de Diciembre de 2012
- [4] http://en.wikipedia.org/wiki/Notification_system, accedido el 4 de Febrero de 2013.
- [6] http://es.wikipedia.org/wiki/Experiencia_de_usuario, accedido el 20 de Octubre de 2012.
- [8] <http://www.nngroup.com/about-user-experience-definition/>, accedido el 16 de Febrero de 2013.
- [9] <http://www.nngroup.com/articles/usability-101-introduction-to-usability/>, accedido el 20 de Noviembre de 2012.

- [10] http://es.wikipedia.org/wiki/Dise%C3%B1o_centrado_en_el_usuario
- [11] http://www.aipo.es/definicion_aipo.php
- [12] <http://www.nngroup.com/about/userexperience.html>, accedido en Noviembre 2012.

[13], ya que es el más conocido y usado, que ofrece sus estadísticas de forma libre. Los resultados son los siguientes:



Ilustración 1: gráfica evolutiva uso SO Móviles

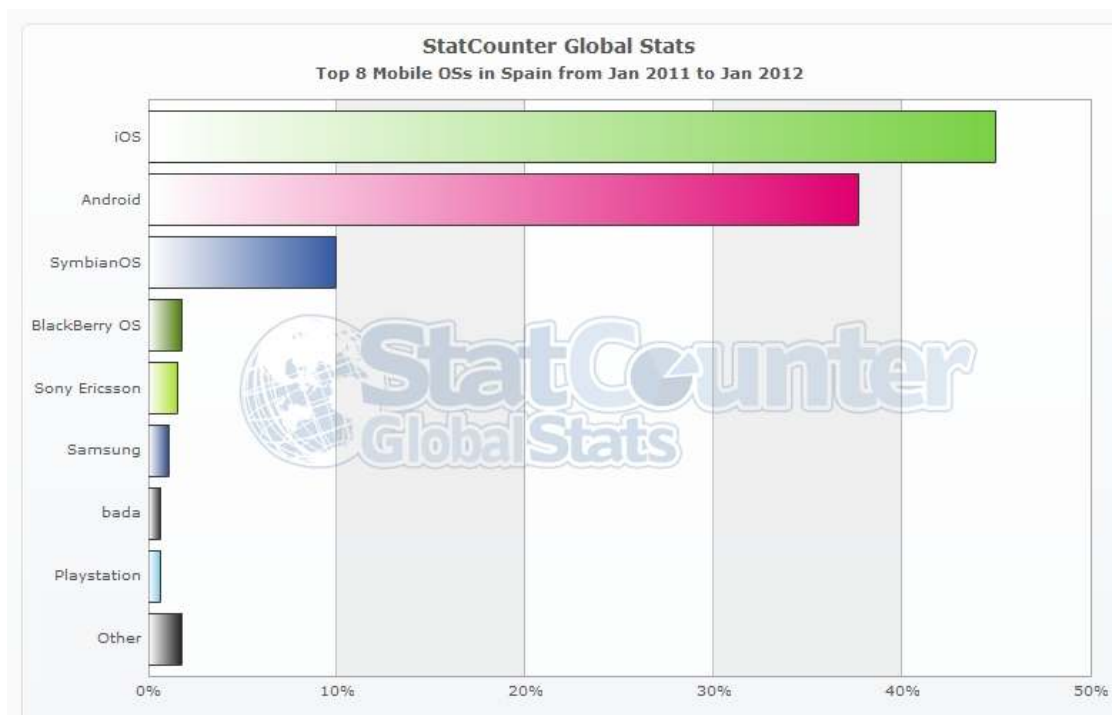


Ilustración 2: gráfica uso total SO móviles

En las Ilustraciones 1 y 2 se ven las estadísticas generales y en evolución del último año de S.O. móviles en España.

Se puede ver (gráfico superior) que Android es en la actualidad el S.O. con mayor uso en nuestro país, seguido de cerca por iOS. La siguiente alternativa sería SymbianOS, pero en el gráfico evolutivo se muestra que sigue una tendencia decreciente en su uso a lo largo del último año.

Ventas

Las estadísticas de ventas indican el número de dispositivos vendidos con cada S.O., los datos son obtenidos de Gartner[14] y se tratan de estadísticas a nivel mundial en el tercer trimestre de los años 2010 y 2011:

**Worldwide Smartphone Sales to End Users by Operating System in 3Q11
(Thousands of Units)**

Operating System	3Q11 Units	3Q11 Market Share (%)	3Q10 Units	3Q10 Market Share (%)
Android	60,490.4	52.5	20,544.0	25.3
Symbian	19,500.1	16.9	29,480.1	36.3
iOS	17,295.3	15.0	13,484.4	16.6
Research In Motion	12,701.1	11.0	12,508.3	15.4
Bada	2,478.5	2.2	920.6	1.1
Microsoft	1,701.9	1.5	2,203.9	2.7
Others	1,018.1	0.9	1,991.3	2.5
Total	115,185.4	100	81,132.6	100

Source: Gartner (November 2011)

Ilustración 3: Tabla de ventas dispositivos móviles por S.O.

En la Ilustración 3 se puede ver que Android vuelve a dominar el mercado. En este caso, domina en número de unidades vendidas con su S.O. y la evolución con respecto al año anterior es de tres veces en cantidad y el doble en porcentaje del total de dispositivos vendidos. El resto de alternativas vuelven a ser Symbian, iOS, RIM (BlackBerryOS) y Microsoft (Windows Mobile).

Número de aplicaciones

Las estadísticas de número de aplicaciones, en cantidad y por descargas, pueden ayudar a hacerse una idea sobre la calidad de los mercados de aplicaciones y de los usuarios de las distintas plataformas, los datos obtenidos[15] son:

	Android Market	Apple Store	Windows Marketplace	App World
Número de aplicación en Market	400.000	550.000	61.000	43.000
Aplicaciones descargadas	10.000.000.000	25.000.000.000	-	-

Tabla 1: Tabla comparativa aplicaciones por S.O.

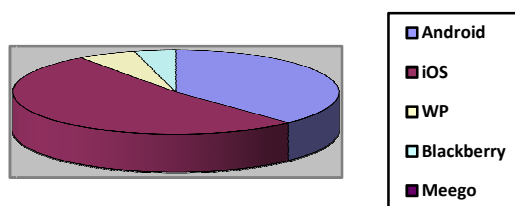


Ilustración 4: gráfico número de aplicaciones

La Ilustración 4 refleja que el mercado con mayor número de aplicaciones y con mayor cantidad de descargas es Apple Store, seguida de Android Market, mientras que los datos de Windows Marketplace (Windows Phone) y App World (BlackBerry OS) están bastante alejados.

Vistos los datos sobre uso, ventas y aplicaciones se ha decidido realizar el estudio de Android, IOS, BlackBerry OS y Windows Phone. Son aquellos S.O. cuya tendencia de crecimiento en uso, número de terminales vendidos y de aplicaciones creadas fue ascendente en el último año.

Android[16]

Es un Sistema Operativo móvil basado en Linux, desarrollado por la Open Handset Alliance (OHA) que fue lanzado inicialmente en Octubre de 2008 y que desde entonces ha sacado múltiples versiones.

El código de las diferentes versiones es liberado bajo dos licencias diferentes: GNU GPL2 para aquellas partes basadas en sistemas que liberasen bajo esa licencia (como Linux) y Apache 2.0 para el código propio.

Una de las características más importantes de Android es su SDK, que permite desarrollar aplicaciones de forma sencilla y que, además, es multiplataforma (Linux, MacOS y Windows). Hay versiones diferentes del SDK para cada versión del SO que sale, siendo liberadas al poco tiempo de ser anunciadas para que los desarrolladores puedan trabajar con ellas.

La distribución de las aplicaciones puede ser realizada de forma directa a través de un paquete en un formato apk o a través de los diferentes mercados de aplicaciones existentes, siendo el más común el Android Market.

iOS[17]

Es un Sistema Operativo móvil desarrollado por Apple basado en su propio sistema operativo para escritorio Mac OS X. El lanzamiento inicial fue en Junio de 2007. Desde entonces ha habido un total de 5 versiones.

El código de las versiones no es liberado por Apple, siguiendo la misma dinámica que sus SO de escritorio, que es completamente cerrado, pero ampliamente optimizado para su propio Hardware, que también es cerrado.

Dispone de un SDK, que debe ser adquirido previo pago y que debe ser utilizado en plataformas basadas en Mac OS. Dicho SDK está pensado para ser programado en Objective-C, utilizando Xcode como IDE.

La distribución de las aplicaciones debe ser realizada a través de la App Store.

Windows Phone[18]

Es un Sistema Operativo móvil desarrollado por Microsoft y basado en su anterior versión de SO móvil: Windows Mobile. El lanzamiento inicial fue en Octubre de 2009, en el que la antigua plataforma Windows Mobile en su versión 6.5, pasó a denominarse Windows Phone. La primera versión lanzada directamente bajo Windows Phone fue la versión 7, anunciada en Febrero de 2010 y lanzada en Octubre de ese mismo año.

El código de las distintas versiones no es liberado por Microsoft, que sigue al igual que en sus SO de escritorio, un modelo de desarrollo basado en Software propietario.

Dispone de un SDK que puede ser descargado de forma gratuita y que permite desarrollar aplicaciones o juegos utilizando Visual Studio, por lo que inicialmente está pensado para utilizarse únicamente en SO Windows.

La distribución de las aplicaciones debe hacerse a través del Windows Marketplace.

Blackberry OS^[19]

Es un Sistema Operativo móvil desarrollado por RIM para sus propios dispositivos, fue lanzado inicialmente en Enero de 1999.

El código de las diferentes versiones no es liberado, por lo que sigue un desarrollo basado en Software propietario.

Dispone de un SDK para el desarrollo de aplicaciones, que puede ser basado en distintas tecnologías: nativas (C, C++), Java, Web (HTML5), dependiendo de la tecnología, puede utilizarse unas herramientas u otras.

La distribución de las aplicaciones debe hacerse a través del App World.

Comparativa

A continuación se presenta una comparativa en la que se estudian diferentes características para cada una de las plataformas seleccionadas importantes para elegir sobre cual desarrollar.

Las características que se van a estudiar son:

Ultima versión: Número y fecha de salida de la última hasta la fecha.

Lanzamiento inicial: fecha del lanzamiento de la primera versión del SO

Libre: indicar si se trata de Software Libre y de código abierto (FLOSS por sus siglas en inglés)

SDK/SDK Libre: Se indica si se ofrece SDK y si es libre.

Multiplataforma: se indica si se puede desarrollar en varias plataformas, típicamente en Windows, Linux y MacOS.

Lenguaje SDK: lenguaje para programar utilizando el SDK.

Herramientas disponibles para IDEs: indicar si junto con el SDK se libera alguna herramienta que ayude al desarrollo para algún IDE, así como el nombre del mismo.

Disponible NDK: se indica si existe algún lenguaje nativo con el que poder programar aplicaciones y si se encuentra documentado.

Sistema de notificaciones integrado: se indica si se dispone de un sistema de gestión de notificaciones integrado en la interfaz principal del SO.

Comunicación integral entre aplicaciones: se indica si el SDK incluye algún sistema de gestión de comunicación entre aplicaciones.

Gestión de persistencia de datos: se indica si se incluye algún motor de base de datos para la persistencia de datos, así como funciones para su fácil acceso.

Multi-Tarea: se indica si el SO permite multitarea

Soporte Multi-Táctil: Se indica si el SO permite el control de interfaces multi-táctiles para acciones, gestures, etc.

3G: se indica si el SO dispone de control de comunicaciones vía 3G y si es posible gestionar a través del SDK.

WiFi: se indica si el SO dispone de control de comunicaciones vía Wifi y si es posible gestionar a través del SDK.

Bluetooth: se indica si el SO dispone de control de comunicaciones vía Bluetooth y si es posible gestionar a través del SDK.

GPS: se indica si el SO dispone de control de posicionamiento vía GPS y si es posible gestionar a través del SDK.

	Característica	Plataforma			
		Android	IOS	Windows Phone	BB OS
Estado	Última Versión	4.0.3 (Diciembre 2011)	5 (Octubre 2011)	7.10 (Sept. 2011)	7 (Agosto 2011)
	Lanzamiento Inicial	Octubre de 2008	Junio de 2007	Octubre de 2009	Enero de 1999
Desarrollo	Software libre	Si	No	No	No
	SDK/SDK Libre	Si/Si	Si/No	Si/No	Si/No
	Multiplataforma	Si	No	No	Si
	Lenguaje SDK	Java	Objective-C	Silverlight(C++/C#), XNA	Java, HTML5, C/C++
	Herramienta para IDE	Si (eclipse)	Si (Xcode)	Si (Visual Studio)	Si (Jdeveloper)
	NDK	Si	No	No	No
Sistema	Sistema de notificación integrado	Si	Si (5.0)	No	Si
	Comunicación entre aplicaciones	Si (Broadcast Receivers)	Si (URLs Schemes)	Si (WCF)	No
	Gestión de persistencia de datos	Si (SQLite, Preferences, File)	Si (SQLite, files)	Si (Sql Server Compact)	Si
	MultiTarea	Si	Si (v4+)	Limitado	Limitado
	Soporte Multitactil	Si	Si	Si	Si
Conectividad	3G	Si	Si	Si	Si
	WiFi	Si	Si	Si	Si
	Bluetooth	Si	Si	No	Si
	GPS	Si	Si	Si	Si

Tabla 2: comparativa características/S.O.

Conclusión

Finalmente se elige Android como Sistema Operativo Móvil sobre el que desarrollar, debido a que tiene unos mejores resultados de uso y ventas con respecto al resto de alternativas y a que ofrece todas las características buscadas. Sobre todo cabe destacar que es libre y multiplataforma, lo que permite que se pueda desarrollar más fácilmente y sin coste alguno.

Además, se ha decidido que se utilizará la versión de la API del SDK más baja que disponga de todas las características estudiadas para poder llegar al mayor número de usuarios posibles, por lo que se desarrollará con la versión 1.5 (SDK API 3) de Android.

2.2 Aplicaciones de gestión y notificación de eventos

En este apartado se estudian una serie de aplicaciones que permiten la gestión y notificación de ciertos eventos o situaciones, el objetivo es obtener una serie de características de estas aplicaciones que sean ideales para una aplicación de gestión de notificaciones.

Selección de aplicaciones

Las aplicaciones que se han seleccionado han sido:

Google Calendar (oficial): Se ha seleccionado esta aplicación porque tiene un sistema de notificaciones de eventos y porque al ser propia de Google, se entiende que el formato de notificaciones y gestión de preferencias podría ser el recomendado para una nueva aplicación.

Gmail (oficial): Se ha seleccionado esta aplicación porque tiene un sistema de notificaciones de eventos y porque al ser propia de Google, se entiende que el formato de notificaciones y gestión de preferencias podría el recomendado para una nueva aplicación.

Facebook (oficial): Se ha seleccionado esta aplicación porque facebook es una aplicación muy extendida, que dispone de un gran equipo para hacer las cosas correctamente en cuanto a la gestión de datos y notificaciones y porque dispone de un API de comunicación con sus sistemas que permite el acceso a sus datos.

TweetCaster (cliente twitter, no oficial): Se ha seleccionado esta aplicación debido a que es el único cliente no oficial disponible para dispositivos con la versión 1.5 de Android.

Google Calendar[20]

Se trata de la aplicación de gestión de calendarios y eventos de Google para móviles, viene instalada por defecto en los dispositivos Android como aplicación nativa.

La aplicación cuenta con una gestión de notificaciones de los eventos al cumplir una determinada fecha (que puede ser configurado), permite seleccionar el tipo de notificaciones, pudiendo ser a través de la barra de notificaciones del sistema o a través de una ventana emergente y permite incluir vibración y un tono específico para las notificaciones.

Los datos de eventos y calendarios son sincronizados en la nube de Google con los datos del usuario configurados en el sistema, permitiendo enviar y recibir datos de este sistema.

Los datos de preferencias y configuración son almacenados en el dispositivo para su acceso rápido.

Gmail[21]

Se trata de la aplicación de gestión de correo electrónico de Google para móviles que viene instalada por defecto en los dispositivos Android como aplicación nativa.

La aplicación cuenta con una gestión de notificaciones de recepción de nuevos correos, permitiendo configurar cada cuanto se quiere hacer la comprobación de nuevos correos, permite configurar si se quiere realizar la notificación a través de la barra de notificaciones del sistema. Además permite incluir vibración y un tono específico para las notificaciones.

Los datos correos electrónicos y etiquetas de los correos son sincronizados en la nube de Google con los datos del usuario configurados en el sistema, permitiendo enviar y recibir datos de este sistema.

Permite abrir otras aplicaciones, como el navegador, youtube o el GPS, si se encuentran enlaces a los distintos formatos dentro de los correos.

Los datos de preferencias y configuración son almacenados en el dispositivo para su acceso rápido.

Facebook[22]

Se trata de la aplicación oficial de la red social facebook para dispositivos móviles Android.

La aplicación cuenta con una gestión de notificaciones muy configurable, pudiendo seleccionar si se quiere o no notificar. Dentro de éstas, se puede elegir configurar: los mensajes, las solicitudes de amistad y la invitación a eventos. Además, permite configurar si se quiere incluir vibración, iluminación del Led (para los móviles que disponen de led), así como el tono de notificación.

Los datos de la aplicación móviles son sincronizados a través de su API con los datos de la aplicación web, guardando los datos locales para acceder rápidamente a ellos.

Permite abrir otras aplicaciones, como el navegador, youtube, fotos, si se encuentran enlaces en los distintos formatos de dichas aplicaciones.

TweetCaster[23]

Se trata de un cliente no oficial de la red social twitter para dispositivos móviles Android.

La aplicación cuenta con una gestión de notificaciones muy configurables, que incluye la notificación cuando llegan nuevos tweets, cuando se reciben nuevas menciones o cuando se reciben nuevos mensajes directos. Además, permite usar vibración, la iluminación LED y un sonido personalizado en las notificaciones.

Los datos de la aplicación móvil son sincronizados a través del API pública de twitter, que permite el acceso a todos los datos de un usuario a las aplicaciones que se permitan. De esta forma, permite obtener los tweets del usuario, escribir un nuevo tweet, realizar búsquedas, recibir menciones o mensajes directos, ver y marcar favoritos y ver, crear y modificar listas.

Además permite abrir otras aplicaciones, como el navegador, youtube, aplicaciones de gestión de fotos si se encuentran enlaces en los formatos de dichas aplicaciones.

Los datos de preferencias y configuración, así como aquellos obtenidos previamente a través del API son guardados en local para acceder más rápidamente.

Selección de Características

A partir de las aplicaciones estudiadas se obtiene el siguiente listado de características a requerir en una aplicación de este tipo:

Notificación de nuevos eventos/sucesos importantes para la aplicación: comprobar que la aplicación permite la notificación a través del sistema de eventos o sucesos de los que se quiera informar al usuario: nuevos emails, fecha de evento cercana, nuevo tweet,...

Acceso a datos de un sistema externo: se indica si se obtienen datos de un sistema (a través de un API o similar) externo a la aplicación.

Envío de datos a un sistema externo: se indica si se envían datos a un sistema (a través de un API o similar) externo a la aplicación.

Configuración de frecuencia de obtención de notificaciones: se indica si se añaden preferencias para la configuración de la frecuencia en la que se realiza el acceso a los datos para poder notificar. Depende del tipo de aplicación variará de minutos a días.

Configuración de tipo de notificación: se indica si se añaden preferencias para la configuración de cómo notifica la aplicación al usuario: email, barra de notificaciones, aviso emergente, etc.

Almacenamiento de datos en local: se indica si se realiza almacenamiento de datos en local para cachear ciertos datos de acceso común.

Comunicación con otras aplicaciones: se indica si se realiza comunicación hacia otras aplicaciones para realizar ciertas acciones, tales como abrir vídeos en una aplicación específica, enlaces en un navegador o similar.

Resumen

A continuación se muestra una tabla comparativa con las características estudiadas para cada aplicación, indicando si dispone o no de ella.

Característica	Aplicación			
	Google Calendar	Gmail	Facebook	Tweetcaster
Notificación	Si	Si	Si	Si
Acceso Sistemas Externos	Si	Si	Si	Si
Envío a Sistemas Externos	Si	Si	Si	Si
Configuración frecuencia notificaciones	Si	Si	Si	Si
Configuración tipo notificación.	Si	Si	Si	Si
Almacenamiento local	Si	Si	Si	Si
Comunicación otras apps.	No	Si	Si	Si

Tabla3: comparativacaracterísticas/aplicación

Conclusiones

Se considera que todas las características estudiadas son importantes para un sistema que gestione y notifique eventos, pero las más importantes entre ellas son: la posibilidad de notificar, la posibilidad de configurar las notificaciones y la comunicación para obtener y enviar información a sistemas

3 Gestión de proyecto software

La idea de este proyecto es realizar una aplicación para dispositivos móviles Android que permita la notificación de la emisión de capítulos de series de televisión para lo que debe ser capaz de buscar y gestionar datos de series y de sus episodios.

Además, debe permitir que los usuarios reciban información sobre la valoración de las series y los episodios, así como valorar los propios episodios para ayudar a que seleccionen que series elegir como favoritas y qué episodios desean ver.

Los datos de la aplicación serán obtenidos a través de un conjunto de servicios web, algunos de los cuales serán necesarios implementar como capa intermedia al no estar disponible de forma directa.

3.1 Plan de trabajo

Identificación de tareas

El proyecto, como cualquier proyecto de software, consta de distintas tareas englobadas dentro de dos grandes bloques: Gestión de proyecto y desarrollo de software.

Dentro del bloque de gestión de proyecto, se encuentran las siguientes tareas:

- **Estudio de alternativas:** análisis de las alternativas de plataformas y aplicaciones para el problema planteado para el proyecto.
- **Análisis:** análisis del proyecto y de sus requisitos. A su vez se dividirá en:
 - Definición de requisitos.
 - Especificación de los requisitos.
- **Diseño:** diseño de la solución. Donde se encuentran las tareas:
 - Diseño del modelo de datos.
 - Diseño de la capa de servicios de acceso a los datos.
 - Diseño de la interfaz de usuario.
 - Diseño de la lógica de negocio.
- **Documentación:** poner todo el conocimiento del proceso completo de desarrollo de software por escrito. Se divide en:
 - Documentación de la fase inicial.
 - Documentación de la implementación y las pruebas.
 - Manual de usuario.

Dentro del bloque de desarrollo de software se encuentran las tareas:

- **Implementación:** dar forma al diseño de la solución planteado, programando y construyendo los sistemas necesarios para su funcionamiento. Se divide en:
 - Implementación de capa intermedia de acceso a datos.
 - Implementación de lógica de negocio.
 - Desarrollo de la interfaz de usuario.
- **Pruebas:** realización de las pruebas automáticas y manuales que dan conformidad a los requisitos planteados en la fase de análisis. Se divide en:
 - Pruebas unitarias.
 - Pruebas funcionales.
 - Pruebas de usabilidad.

Estimación de tareas

A continuación se muestra la estimación en días de cada una de las tareas identificadas, hay que tener en cuenta que se consideran días en jornadas de 8 horas.

Tarea	Sub-Tarea	Estimación
Estudio de alternativas	Estudio de plataformas	3 días
	Estudio de aplicaciones	3 días
	Total	6 días
Análisis	Definición de requisitos	5 días
	Especificación de requisitos	10 días
	Total	15 días
Diseño	Diseño del modelo de datos	3 días
	Diseño de la capa de servicios	7 días
	Diseño de la lógica de negocio	7 días
	Diseño de la interfaz de usuario	8 días
	Total	25 días
Documentación	Documentación fase inicial	7 días
	Documentación de implementación y pruebas	8 días
	Total	15 días
Implementación	Implementación de capa de servicios	10 días
	Implementación de lógica de negocio	15 días
	Implementación de interfaz de usuario	15 días
	Total	40 días
Pruebas	Pruebas unitarias	5 días
	Pruebas funcionales	5 días
	Pruebas de usabilidad	5 días
	Total	15 días
TOTAL		116 días

Tabla 4: estimación de las tareas

El total de días estimados para el proyecto sería de 116 días en un proceso secuencial.

3.2 Gestión de recursos

En este apartado se detallan los recursos necesarios para la realización del proyecto, donde se incluyen tanto recursos personales como los recursos materiales.

Además se realizará una asignación de los recursos personales en las distintas tareas especificadas en el apartado 3.2 que ayudará a obtener un coste estimado del proyecto que sumado a los costes materiales, darán el presupuesto total del proyecto.

Especificación de recursos

Recursos humanos

El desarrollo del proyecto requiere de un equipo personal cualificado que pueda realizar las tareas en un tiempo razonable con una buena calidad y que cumplan las necesidades del problema.

Para todo ello se requieren los siguientes roles dentro del equipo:

- **Jefe de proyecto:** encargado de gestionar el trabajo de todo el equipo, dividir y gestionar tareas y ayudar a resolver dudas y problemas al resto de expertos. Además debe encargarse de documentar todos los procesos llevados a cabo en el proyecto.
- **Analista:** encargado de realizar el análisis del proyecto, estudiar las necesidades y convertirlas en requisitos tangibles que puedan ser implementados y probados.
- **Arquitecto Software:** encargado de realizar el diseño de la solución y de dar soporte técnico al resto del equipo técnico.
- **Desarrollador Web:** encargado de desarrollar la lógica de negocio y la capa de acceso a datos de la parte web de la aplicación.
- **Desarrollador Móvil:** encargado de desarrollar la lógica de negocio y de presentación de la parte móvil de la aplicación.
- **Gestor de calidad:** encargado de diseñar y realizar las pruebas unitarias y funcionales que aseguren el cumplimiento de los requisitos del proyecto.
- **Experto en usabilidad:** encargado de realizar los prototipos y diseños de la aplicación para mejorar la experiencia de los usuarios, así como de realizar los tests de usabilidad y dar a conocer los resultados para realizar las modificaciones pertinentes.

Dado que el proyecto no tiene un tamaño excesivo, contar con una persona para cada rol indicado podría aumentar los costes del proyecto demasiado y complicar y retrasar la realización del mismo, por ello se considera la inclusión de equipo multidisciplinar que permita reducir al mínimo el número de integrantes, con lo que finalmente se dispondría de un equipo de cuatro personas, con los siguientes participantes:

- 1 jefe de proyecto que hará las tareas de Analista en las fases iniciales del proyecto.
- 1 arquitecto de software que se encargue a su vez de diseñar las pruebas y asegurar el cumplimiento de las mismas.
- 1 desarrollador multidisciplinar que desarrolle tanto la parte web como la parte móvil.
- 1 experto en usabilidad que se encargue de prototipos, diseño y gestión de pruebas de usabilidad.

Recursos materiales

Para la realización del proyecto es necesario contar con una serie de materiales físicos o digitales que permitan realizar todas las tareas definidas de forma óptima, son los siguientes:

- Ordenadores personales con los que realizar los trabajos del día a día del proyecto:
 - 4 Dell OptiPlex 580 para el resto del equipo: 449€.
- 1 servidor donde se alojará los componentes web de la aplicación:
 - Dell Power Edge T310: 649€.
- 1 impresora para poder imprimir los diferentes documentos del proyecto por parte de los miembros del equipo:
 - Impresora multifunción HP Officejet 4500 Wireless: 89€.
- Licencias de Windows 7 Professional (incluido en Dell) con los que realizar la documentación del proyecto.
- 4 licencias de Microsoft Office para que los componentes del equipo puedan documentar los diferentes procesos que lleven a cabo
 - 3 licencias 2007 Professional : 699€
 - 1 licencia Office Mac 2010: 249€.
- Microsoft Project para que el jefe de proyecto pueda realizar los documentos de gestión de proyectos, 1 licencia: 999.95€.
- Dispositivos móviles con los que realizar las diferentes pruebas de la aplicación.
 - 1 Motorola Dext (249€) y 1 Samsung Galaxy SCL (309€).

Asignación de recursos

Los diferentes recursos personales que se encargarán de realizar el proyecto deben ser asignados a las tareas para poder realizarlas de la manera más óptima cumpliendo las dependencias que existen entre ellas tal y como se detalla en el punto de planificación de las tareas.

Para ello se presenta tabla5 en la que se muestra para cada una de las tareas de primer nivel definidas en el punto 3.2, los responsables de dicha tarea que se corresponden con el conjunto de roles de cada uno de los miembros del equipo indicados en el punto Recursos humanos:

Tarea	Sub-Tarea	Responsable
Estudio de alternativas	Estudio de plataformas	Jefe de proyecto, Arquitecto SW
	Estudio de aplicaciones	Jefe de proyecto
Análisis	Definición de requisitos	Jefe de proyecto
	Especificación de requisitos	Jefe de proyecto
Diseño	Diseño del modelo de datos	Arquitecto SW
	Diseño de la capa de servicios	Arquitecto SW, Desarrollador
	Diseño de la lógica de negocio	Arquitecto SW, Desarrollador
	Diseño de la interfaz de usuario	Experto usabilidad
Documentación	Documentación fase inicial	Jefe de proyecto
	Documentación implementación y pruebas	Arquitecto SW , Desarrollador y experto usabilidad
Implementación	Implementación de capa de servicios	Desarrollador
	Implementación de lógica de negocio	Desarrollador
	Implementación de interfaz de usuario	Desarrollador, Experto en usabilidad
Pruebas	Pruebas unitarias	Arquitecto SW
	Pruebas funcionales	Arquitecto SW
	Pruebas de usabilidad	Experto usabilidad

Tabla 5: asignación de recursos a tareas

Con ello podemos ver que la dedicación en hombres/día para cada uno de los participantes del equipo sería como sigue:

Recurso	Días dedicados
Jefe de proyecto	26.5
Arquitecto SW	24.17
Desarrollador	42.17
Diseñador	23.17

Tabla 6: dedicación en días

Que se calcularía de sumar las tareas que tiene asignada, dividiendo entre el número de participantes aquellas con más de un participante.

3.3 Presupuesto

En base a la planificación de tareas y recursos de los apartados 3.2 y 3.3, se presenta a continuación el presupuesto estimado del proyecto, en el que se incluyen los gastos de personal, los gastos de equipos y otros gastos asociados para finalmente presentar un presupuesto general.

Costes de personal

La siguiente tabla muestra los gastos de personal basados en el cálculo del coste real de cada tipo de recurso, siendo el coste por mes de 2.694,39 € por Ingeniero, rango en el que se encuentran el desarrollador y el experto en usabilidad y de 4.289,54 € por Ingeniero Sénior, rango en el que se encuentran el Jefe de proyecto y el Arquitecto Software.

Por tanto el gasto de personal para el proyecto siendo el tiempo de hombre mes 131.25 horas es:

Recurso	Dedicación (hombre/mes)	Coste hombre/mes	Coste (€)
Jefe de proyecto	1.61	4289.54	6906.16
Arquitecto SW	1.47	4289.54	6305.62
Desarrollador	2.57	2694.39	6924.58
Diseñador	1.41	2694.39	3799.09
TOTAL			23935.46

Tabla 7: coste personal

Costes de recursos materiales

El coste total de los recursos materiales basados en los recursos indicados en el apartado 3.3.1.2 Recursos materiales sería el siguiente:

Descripción	Coste (€)	% Uso dedicado proyecto	Dedicación (meses)	Periodo de depreciación	Coste imputable (€)
Dell OptiPlex 580	449.00	100	1.61	60	12.05
Dell OptiPlex 580	449.00	100	1.47	60	11.00
Dell OptiPlex 580	449.00	100	2.57	60	19.23
Dell OptiPlex 580	449.00	100	1.41	60	10.55
Dell Power EdgeT310	649	100	12	60	129.80
Impresora HP Officejet 4500	89.00	100	7.06	60	10.47
Microsoft Office Professional	699	100	1	60	11.65
Microsoft Office Professional	699	100	1	60	11.65
Microsoft Office Professional	699	100	1	60	11.65
Microsoft Project	999.95	100	7.06	60	117.66
Motorola Dext	249	100	5	60	20.75
Samsung SCL	309	100	5	60	25.75
TOTAL					392.21

Tabla 8: coste de recursos materiales

Dicho coste se obtiene de calcular el coste de cada producto, por su tiempo de uso (en meses) dividido por el periodo de depreciación de los productos.

El tiempo de uso está basado en las tareas que requieren dicho recurso en el que serán utilizados por los miembros del equipo.

Coste total

Con lo visto en las tablas 7 y 8, el coste del proyecto estimado sería:

Descripción	Coste imputable (€)
Personal	23935.46
Recursos materiales	392.21
TOTAL	24327.67

Tabla 9: coste global del proyecto

A lo que habría que sumar el beneficio esperado y los costes estimados de riesgos, que pretenden reducir el perjuicio por posibles contratiempos en el desarrollo del proyecto:

Descripción	Porcentaje	Coste imputable (€)
Coste		24327.67
Beneficio	20%	4865.53
	Total	29193.20
Riesgo	5%	1459.66
	TOTAL	30652.86

Tabla 10: coste post beneficio y riesgo

A lo que habría que aplicar el IVA del 18%, por lo que el valor total del proyecto sería:

Descripción	Porcentaje	Valor
Coste		30652.86
IVA	21%	6437.10
	TOTAL	37089.96

Tabla 11: Coste final del proyecto

El presupuesto total de este proyecto asciende a la cantidad de **treinta y siete mil ochenta y nueve con noventa y seis euros (36955.18€)**.

4 Solución

En este apartado se explicará de forma detallada la solución llevada a cabo para resolver el problema descrito en los anteriores puntos del documento. El detalle de la solución se centrará en el proceso de desarrollo seguido para llegar a la solución.

4.1 Descripción de la solución

Para poder solventar la problemática planteada y poder disponer de una herramienta de notificación de emisión de series de forma ubicua y sencilla se va a realizar una aplicación para dispositivos móviles Android que permita la notificación de la emisión de capítulos de series de televisión.

La aplicación debe ser capaz de obtener la información necesaria a través de servicios, debido a la falta de servicios web actuales que nos proporcionen los datos requeridos, se proveerá un servicio web propio que sirva de filtro. Dicho servicio web será actualizado de forma periódica

4.2 El proceso de desarrollo

En este punto se detalla el proceso de desarrollo de Software de la solución propuesta. Se encuentra dividido en los siguientes apartados:

- Análisis
- Diseño
- Implementación
- Pruebas

Análisis

En este apartado se detallan las características y objetivos del sistema a desarrollar, las necesidades que debe cubrir, analizando y clasificando dichas necesidades.

Definición de requisitos

En esta sección se detallan los requisitos de usuario de la aplicación, que se dividen en:

- Requisitos funcionales: funcionalidad que la aplicación debe proveer para cumplir sus objetivos.
- Requisitos no funciones: especificación de cómo se debe construir la aplicación, que se requiere para que funcione y como debe responder.

Cada tipo de requisito se encuentra especificado en una tabla aparte y para cada requisito se especifica los siguientes valores:

- Identificador: valor identificativo del requisito, único. Sigue el formato: RU-[NF|F]XXX.
- Necesidad: importancia en el cumplimiento del requisito para la viabilidad de la aplicación, pudiendo ser esencial, deseable u opcional.
- Prioridad: orden en el que debe cumplirse los requisitos, pudiendo ser alta, media o baja.
- Descripción: breve explicación del requisito.

Requisitos Funcionales

Identificador	Necesidad	Prioridad	Descripción
RU-F001	Esencial	Alta	La aplicación deberá permitir realizar una búsqueda de series por nombre.
RU-F002	Esencial	Alta	La aplicación deberá permitir ver la ficha de una serie, que incluirá información detallada de dicha serie.
RU-F003	Esencial	Alta	La aplicación deberá permitir marcar una serie como favorita.
RU-F004	Esencial	Alta	La aplicación deberá permitir ver el listado de series favoritas.
RU-F005	Esencial	Alta	La aplicación deberá permitir desmarcar una serie como favorita.
RU-F006	Opcional	Baja	La aplicación deberá mostrar un listado con las series mejor valoradas.
RU-F007	Esencial	Alta	La aplicación deberá notificar la emisión de nuevos episodios de las series favoritas.
RU-F008	Esencial	Alta	La aplicación deberá permitir que el usuario marque episodios como vistos.
RU-F009	Deseable	Media	La aplicación deberá permitir votar los episodios vistos.
RU-F010	Deseable	Media	La aplicación deberá permitir configurar la frecuencia de comprobación para las notificaciones.
RU-F011	Deseable	Media	La aplicación deberá permitir configurar la hora de las comprobaciones para las notificaciones.
RU-F012	Deseable	Media	La aplicación deberá guardar un histórico de los episodios emitidos.
RU-F013	Opcional	Baja	La aplicación deberá permitir configurar el formato de notificación, pudiendo elegir activar notificación de sistema, por email y a través del led.
RU-F014	Esencial	Alta	La aplicación deberá permitir obtener las series que hayan emitido algún episodio nuevo desde una fecha.
RU-F015	Esencial	Alta	La aplicación deberá permitir obtener los episodios que hayan sido emitidos desde una fecha.
RU-F016	Esencial	Alta	La aplicación deberá permitir obtener la puntuación media de las valoraciones de una serie.
RU-F017	Esencial	Alta	La aplicación deberá permitir obtener la puntuación media de un episodio.

Tabla 12: Requisitos Funcionales

Requisitos No Funcionales

Identificador	Necesidad	Prioridad	Descripción
RU-NF001	Esencial	Alta	El cliente móvil de la aplicación deberá ser compatible con Android a partir de la versión 1.5.
RU-NF002	Esencial	Alta	El cliente móvil de la aplicación deberá ser compatible con tamaños de pantalla a partir de 320x480px
RU-NF003	Deseable	Media	El cliente móvil de la aplicación deberá ser compatible con tamaños de pantalla de cómo máximo 480 x 800px.
RU-NF004	Esencial	Alta	El cliente móvil de la aplicación deberá almacenar los datos en la base de datos en local de cada dispositivo (SQLite).
RU-NF005	Deseable	Media	El cliente móvil deberá mostrar un mensaje de acción para todas aquellas acciones que requieran un cierto tiempo de respuesta.
RU-NF006	Esencial	Alta	El cliente móvil deberá mostrar un dialogo de confirmación ante cualquier acción crítica del sistema.
RU-NF007	Deseable	Media	El cliente móvil deberá mostrar un mensaje confirmando la realización de las acciones.
RU-NF008	Esencial	Alta	La aplicación estará basada en una Arquitectura Basada en Servicios (SOA) que darán acceso a la información al cliente móvil.
RU-NF009	Esencial	Alta	La aplicación deberá disponer de un Servidor Web que haga de capa intermedia de acceso a los datos que no provea el servicio web principal.
RU-NF010	Deseable	Media	La aplicación web deberá servir los datos en distintos formatos: XML y JSON.
RU-NF011	Esencial	Alta	El servidor web en el que corra los servicios intermedios de la aplicación deberá disponer de una base de datos MySQL.
RU-NF01/	Deseable	Media	El servidor web en el que corra los servicios intermedios de la aplicación deberá disponer de una instalación de PHP.

Tabla 13: Requisitos no funcionales

Especificación de requisitos

Para la especificación de los requisitos, se describirán y detallarán los casos de uso de la aplicación.

En la siguiente tabla se ven los distintos casos de uso de la aplicación:

Identificador	Nombre	Requisito/s
CU-001	Acceder a la aplicación	
CU-002	Buscar una serie	RU-F001
CU-003	Ver la información de una serie	RU-F002
CU-004	Marcar una serie como favorita	RU-F003
CU-005	Desmarcar una serie como favorita	RU-F005
CU-006	Acceder al listado de las series mejor valoradas	RU-F006
CU-007	Acceder al listado de mis series	RU-F004
CU-008	Acceder al listado de episodios emitidos	RU-F007 y RU-F015
CU-009	Modificar la frecuencia de comprobación	RU-F010
CU-010	Modificar la hora de comprobación	RU-F011
CU-011	Ver las notificaciones sobre nuevos episodios	RU-F007, RU-F014 y RU-F015
CU-012	Marcar un episodio como visto	RU-F008
CU-013	Puntuar un episodio.	RU-F009
CU-014	Ver la nota de una serie	RU-F002, RU-F016
CU-015	Ver la nota de un episodio	RU-F007, RU-F015 y RU-F017

Tabla 14: tabla de casos de uso

A continuación se muestra en detalle cada uno de los casos de uso presentados anteriormente:

Identificador	Nombre	Importancia	Actor
CU-001	Acceso a la aplicación	Alta	Cualquier usuario
Descripción			
El usuario intenta acceder a la aplicación a través de su dispositivo.			
Precondiciones			
Dispositivo móvil con Android 1.5+			
Aplicación instalada.			
Flujo Normal de eventos			
<ol style="list-style-type: none"> 1. El usuario abre el menú de aplicaciones 2. El usuario pulsa sobre el icono de la aplicación 3. La aplicación se abre, mostrando la pantalla inicial. 			
Flujo Alternativo de eventos			
<ol style="list-style-type: none"> 1. El usuario accede al escritorio del dispositivo. 2. El usuario pulsa sobre el icono de aplicación. 3. La aplicación se abre, mostrando la pantalla inicial. 			

Tabla 15: Caso de uso CU-001

Identificador	Nombre	Importancia	Actor
CU-002	Realizar una búsqueda	Alta	Cualquier usuario
Descripción			
El usuario intenta realizar una búsqueda de una serie mediante el buscador.			
Precondiciones			
CU-001			
Flujo Normal de eventos			
<ol style="list-style-type: none"> 1. El usuario introduce un nombre en el campo de búsqueda. 2. El usuario pulsa el botón de buscar. 3. El usuario se encuentra con una lista de resultados de series que coinciden con la búsqueda realizada. 			
Flujo Alternativo de eventos			
<ol style="list-style-type: none"> 1. El usuario introduce un nombre en el campo de búsqueda. 2. El usuario pulsa el botón de buscar. 3. El usuario se encuentra un mensaje indicando que no hay resultados para la búsqueda, así como un campo de texto para introducir una nueva búsqueda. 			
Flujo Alternativo de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa sobre la pestaña de Buscar de la aplicación. 2. El usuario introduce un nombre en el campo de búsqueda. 3. El usuario pulsa el botón de buscar. 4. El usuario se encuentra con una lista de resultados de series que coinciden con la búsqueda realizada. 			

Tabla 16: Caso de uso CU-002

Identificador	Nombre	Importancia	Actor
CU-003	Visualizar la ficha de una serie	Alta	Cualquier usuario
Descripción			
El usuario intenta ver la ficha del detalle de una serie.			
Precondiciones			
CU-002 ó CU-006 ó CU-007			
Flujo Normal de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa sobre un elemento de la lista de series. 2. El usuario se encuentra con la información en detalle de una serie. 			

Tabla 17: Caso de uso CU-003

Identificador	Nombre	Importancia	Actor
CU-004	Marcar una serie como favorita	Alta	Cualquier usuario
Descripción			
El usuario pretende seleccionar una serie como favorita para recibir notificaciones.			
Precondiciones			
CU-002 ó CU-006			
Flujo Normal de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa sobre el icono de estrella al lado del nombre de la serie. 2. El usuario pulsa aceptar al mensaje de verificación de acción. 3. El usuario recibe un mensaje de confirmación a la acción. 			
Flujo Alternativo de eventos			
<ol style="list-style-type: none"> 5. El usuario pulsa sobre el icono de estrella al lado del nombre de la serie. 6. El usuario pulsa cancelar al mensaje de verificación de acción. 7. La serie no se guarda como favorita. 			

Tabla 18: Caso de uso CU-004

Identificador	Nombre	Importancia	Actor
CU-005	Desmarcar una serie como favorita	Alta	Cualquier usuario
Descripción			
El usuario pretende indiciar que una serie favorita deja de serlo y no quiere recibir notificaciones.			
Precondiciones			
CU-002 ó CU-006 ó CU-007			
Flujo Normal de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa sobre el icono de estrella marcada al lado del nombre de la serie. 2. El usuario pulsa aceptar al mensaje de verificación de acción. 3. El usuario recibe un mensaje de confirmación a la eliminación. 			
Flujo Alternativo de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa sobre el icono de estrella marcada al lado del nombre de la serie. 2. El usuario pulsa cancelar al mensaje de verificación de acción. 			

Tabla 19: Caso de uso CU-005

Identificador	Nombre	Importancia	Actor
CU-006	Acceder al listado de las series mejor valoradas	Alta	Cualquier usuario
Descripción			
El usuario pretende acceder al listado que muestra las 10 series mejor valoradas por los usuarios.			
Precondiciones			
CU-001 ó CU-007 ó CU-008			
Flujo Normal de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa la pestaña de Top Series de la aplicación. 2. El usuario ve un listado con las series 10 series mejor valoradas. 			

Tabla 20: Caso de uso CU-006

Identificador	Nombre	Importancia	Actor
CU-007	Acceder al listado de mis series (Favoritas)	Alta	Cualquier usuario
Descripción			
El usuario pretende acceder al listado que muestra las series favoritas del usuario.			
Precondiciones			
CU-001 ó CU-006 ó CU-008			
Flujo Normal de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa la pestaña de Mis Series de la aplicación. 2. El usuario ve un listado con las series marcadas. 			

Tabla 21: Caso de uso CU-007

Identificador	Nombre	Importancia	Actor
CU-008	Acceder al listado de episodios emitidos	Alta	Cualquier usuario
Descripción			
El usuario pretende ver el listado de episodios emitidos recientemente y pendientes de ver.			
Precondiciones			
CU-001 ó CU-006 ó CU-007 ó CU-011			
Flujo Normal de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa sobre la pestaña de Nuevos episodios de la aplicación. 2. El usuario ve la lista de series emitidas pendientes de ver. 			
Flujo Alternativo de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa sobre el mensaje de notificación de nuevos episodios. 2. El usuario ve la lista de series emitidas pendientes de ver. 			

Tabla 22: Caso de uso CU-008

Identificador	Nombre	Importancia	Actor
CU-009	Modificar la frecuencia de comprobación	Alta	Cualquier usuario
Descripción			
El usuario pretende modificar la frecuencia en días en la que se realizar la comprobación de nuevos episodios emitidos.			
Precondiciones			
CU-001 ó CU-006 ó CU-007 ó CU-008			
Flujo Normal de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa sobre el botón de menú de su dispositivo. 2. El usuario pulsa sobre la opción Preferencias del menú emergente. 3. El usuario Pulsa sobre Frec. Comprobaciones en el listado de preferencias 4. El usuario elige una frecuencia en días para las comprobaciones y pulsa ok. 5. El usuario recibe un mensaje de confirmación. 			
Flujo Alternativo de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa sobre el botón de menú de su dispositivo. 2. El usuario pulsa sobre la opción Preferencias del menú emergente. 3. El usuario Pulsa sobre Frec. Comprobaciones en el listado de preferencias 4. El usuario pulsa cancelar. 			

Tabla 23: Caso de uso CU-009

Identificador	Nombre	Importancia	Actor
CU-010	Modificar la hora de las comprobaciones	Alta	Cualquier usuario
Descripción			
El usuario pretende modificar la hora del día a la que se hace la comprobación de nuevos episodios.			
Precondiciones			
CU-001 ó CU-006 ó CU-007 ó CU-008			
Flujo Normal de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa sobre el botón de menú de su dispositivo. 2. El usuario pulsa sobre la opción Preferencias del menú emergente. 3. El usuario Pulsa sobre Hora comprobaciones en el listado de preferencias 4. El usuario elige una frecuencia en días para las comprobaciones y pulsa ok. 5. El usuario recibe un mensaje de confirmación. 			
Flujo Alternativo de eventos			
<ol style="list-style-type: none"> 1. El usuario pulsa sobre el botón de menú de su dispositivo. 2. El usuario pulsa sobre la opción Preferencias del menú emergente. 3. El usuario Pulsa sobre Frec. Comprobaciones en el listado de preferencias 4. El usuario pulsa cancelar. 			

Tabla 24: Caso de uso CU-010

Identificador	Nombre	Importancia	Actor
CU-011	Ver las notificaciones sobre nuevos episodios	Alta	Cualquier usuario
Descripción			
El usuario pretende acceder a la notificación sobre nuevos episodios emitidos.			
Precondiciones			
CU-004			
Flujo Normal de eventos			
<ol style="list-style-type: none">1. El usuario recibe una notificación de la aplicación.2. El usuario pulsa el área de notificaciones del sistema.3. El usuario selecciona la notificación de la aplicación.			

Tabla 25: Caso de uso CU-011

Identificador	Nombre	Importancia	Actor
CU-012	Marcar un episodio como visto	Alta	Cualquier usuario
Descripción			
El usuario pretende marcar un episodio de la lista de pendientes como visto.			
Precondiciones			
CU-008			
Flujo Normal de eventos			
<ol style="list-style-type: none">1. El usuario pulsa sobre el icono al lado del episodio.2. El usuario pulsa ok al mensaje de verificación.3. El usuario recibe un mensaje de confirmación.			
Flujo Alternativo de eventos			
<ol style="list-style-type: none">4. El usuario pulsa sobre el icono al lado del episodio.5. El usuario pulsa cancelar al mensaje de verificación.			

Tabla 26: Caso de uso CU-012

Identificador	Nombre	Importancia	Actor
CU-013	Puntuar un episodio	Alta	Cualquier usuario
Descripción			
El usuario pretende puntuar un episodio visto.			
Precondiciones			
CU-008			
Flujo Normal de eventos			
<ol style="list-style-type: none">1. El usuario pulsa sobre el icono al lado del episodio.2. El usuario introduce un número para valorar y pulsa ok.3. El usuario recibe un mensaje de confirmación.			
Flujo Alternativo de eventos			
<ol style="list-style-type: none">4. El usuario pulsa sobre el icono al lado del episodio.5. El usuario pulsa cancelar en la ventana emergente.			

Tabla 27: Caso de uso CU-013

Identificador	Nombre	Importancia	Actor
CU-014	Ver la nota de una serie.	Alta	Cualquier usuario
Descripción			
El usuario pretende ver la valoración media de una serie.			
Precondiciones			
CU-003			
Flujo Normal de eventos			
1. El usuario observa la nota en la parte superior derecha de la ficha de la serie.			

Tabla 28: Caso de uso CU-014

Identificador	Nombre	Importancia	Actor
CU-015	Ver la nota de un episodio	Alta	Cualquier usuario
Descripción			
El usuario pretende ver la valoración media de un episodio.			
Precondiciones			
CU-008			
Flujo Normal de eventos			
1. El usuario observa la nota en la información de cada episodio del listado.			

Tabla 29: Caso de uso CU-015

Diseño

En este apartado se descompondrá y organizará el sistema en elementos para que posteriormente puedan ser desarrollados por separado.

Diseño de sistema

El sistema a desarrollar se plantea como una aplicación para dispositivos móviles, que será el cliente de la aplicación, y un conjunto de servicios que proveerán de información y funcionalidad al cliente. De entre los servicios habrá algunos que se desarrollarán y otros en los que se usará un API externo que ya provee dichos datos.

En líneas generales la arquitectura de la aplicación sería:

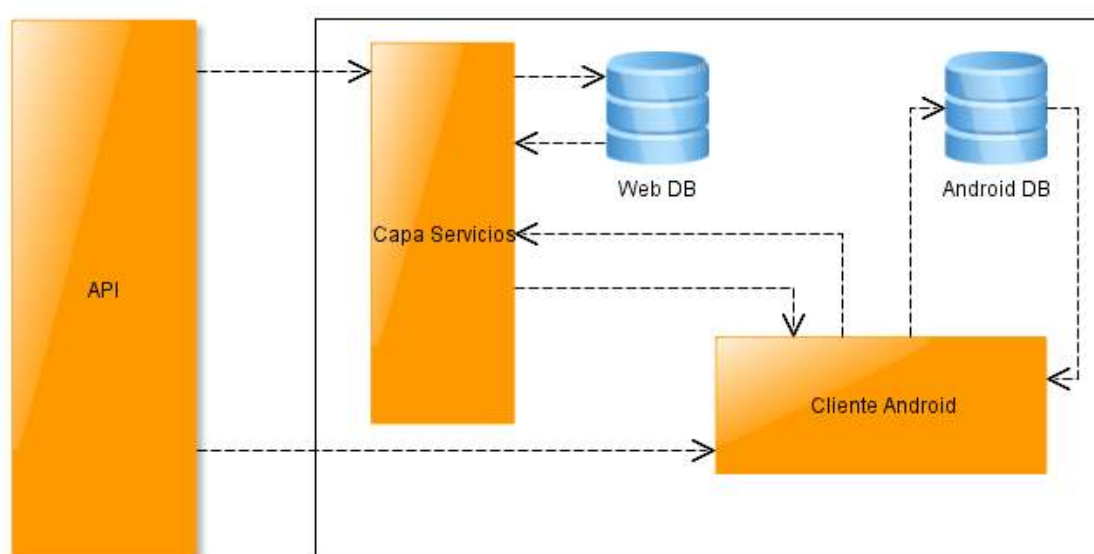


Ilustración 5: esquema general de la aplicación

API externa

El API externa provista a través de thetvdb.com servirá para acceder a la colección de información sobre series y episodios que dicho API dispone a través de servicios Web. Dichos servicios devuelven XML como formato de respuesta.

Sirve para realizar las búsquedas de series, la obtención de información detallada (ficha de serie), así como para poblar de datos y actualizar periódicamente la Capa de Servicios de la aplicación.

Capa de Servicios

La capa de servicios web de la aplicación estará desarrollada utilizando PHP como lenguaje de programación y MySQL como gestor de base de datos. Esta capa intermedia se encargará de proveer aquella funcionalidad que el API externo no pueda proveer, así como de almacenar la información de uso de la aplicación que resulte interesante, como la valoración de episodios de los usuarios.

Está basado en una arquitectura de servicios controlador – modelo, siendo el controlador el gestor de las peticiones y el modelo el que abstrae la lógica hacia la base de datos, obteniendo los datos y ofreciéndoselos al controlador para su devolución correctamente formateados.

Usará El API externo para actualizar periódicamente la información sobre series y episodios y proveerá al cliente de funcionalidad para conocer la valoración de series y episodios, así como de las actualizaciones que se dispongan de las series y episodios en base a una fecha.

Esta capa provee los servicios tanto en XML como en JSON.

Cliente Móvil

El cliente móvil Android estará desarrollado en Java utilizando el SDK disponible para desarrollar en nativo y utilizará SQLite para almacenar los datos que se quieran cachear en el dispositivo.

Dispondrá de un pseudo-modelo que sirve de abstracción a las clases para el acceso a la base de datos.

Las interfaces se desarrollan de forma nativa mezclando layouts y otros componentes basados en XML.

La lógica de negocio y de comunicación entre la aplicación están basadas en Clases Java de tipo Activity.

Usará ambos servicios (API y Capa de Servicios) para acceder a los distintos datos que sirvan dotar de la funcionalidad deseada.

Diseño detallado

Se describe el diseño de las capas de PERSISTENCIA, MODELO e INTERFAZ del sistema.

Persistencia

Web

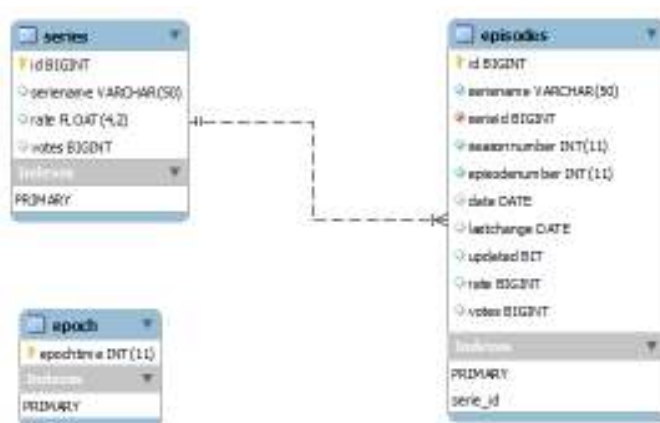


Ilustración 6: Esquema de la base de datos web

- **epoch**: representación de la fecha de la última actualización de datos desde TheTvDB. Se guarda únicamente el valor temporal en TIMESTAMP, la elección de este formato

es porque es el que requiere el API de TheTvDB.

- **series**: representación de la información importante de una serie que incluye nombre, identificador, calificación (rate) y cantidad de votos (votes). El identificador corresponde al usado por TheTvDB.
- **episodes**: representación de la información importante de un episodio, que incluye identificador, nombre de serie, identificador de serie, número de temporada, capítulo dentro de la temporada, así como la fecha de emisión, la fecha de la última actualización en TheTvDB y de la calificación (rate, sumatorio de puntuaciones) y el número de votos (votes). El identificador de serie es una clave foránea sobre el identificador de la tabla series. El identificador corresponde al usado por TheTvDB.

Móvil

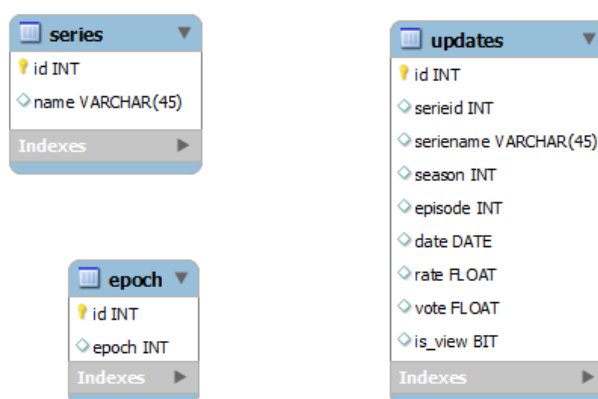


Ilustración 7: Esquema de la base de datos del móvil

- **epoch**: representación de la fecha de la última comprobación de actualizaciones. Se guarda únicamente el valor temporal en `TIMESTAMP`, la elección de este formato es porque es el que requiere el API de TheTvDB.
- **series**: representación de la información importante de las series marcadas por el usuario como favoritas incluye el identificador de la serie y su nombre únicamente.
- **updates**: representación de la información de los episodios que han sido obtenidos como emitidos en las actualizaciones, incluye el identificador del episodio, nombre de serie, identificador de serie, número de temporada, capítulo dentro de la temporada, así como la fecha de emisión, la calificación (rate, votación media), la puntuación realizada por el usuario (si la ha hecho) y un flag que indica si se ha marcado como visto.

Servicios Web

TheTvDB

El API theTheTvDB está basado en XML y ofrece una serie de interfaces a la que acceder para obtener información de series, episodios y banners. Además, gestiona la actualización de los datos de las series, para mantener toda la información al día.

Los distintos métodos del API que interesan para la aplicación son:

- **GetSeries:** obtención de información de serie en base a un nombre.
- **Updates:** obtención de actualizaciones de información de series y episodios.
- **Serie:** obtención de información de una serie basada en un identificador.
- **Episode:** obtención de información de un episodio basado en un identificador.
- **Banners:** obtención de un banner para poder mostrar en la ficha de una serie

A continuación procedemos a explicar en detalle cada uno de los interfaces, así como la forma en la que son usados para guardar los datos en una base de datos propia.

GetSeries

Se trata de una interfaz que permite el acceso a información de series basándose en un parámetro de búsqueda que representa el nombre de la serie.

Se encarga de devolver aquellas series cuyo nombre contiene el texto introducido bajo el parámetro que representa el nombre de la serie.

El formato de petición sería:

<http://www.thetvdb.com/api/GetSeries.php?seriesname=<seriesname>>

Donde se debería sustituir `<seriesname>` por el nombre o parte del nombre de la serie a buscar.

Dicha interfaz nos será muy útil para su uso directamente desde las aplicaciones o clientes, ya que no permite mostrar información de muchas series sin almacenarlas todas en nuestro sistema.

El formato del XML sigue el siguiente DTD:

```
<!DOCTYPEData[
  <!ELEMENTData(Series+)>
  <!ELEMENTSeries(seriesid, language, SeriesName, banner?, Overview?, FirstAired?, IMDB_ID?, zap2it_id?, id)>
  <!ELEMENTseriesid(PCDATA)>
  <!ELEMENTlanguage(PCDATA)>
  <!ELEMENTSeriesName(PCDATA)>
  <!ELEMENTbanner(PCDATA)>
  <!ELEMENTOverview(PCDATA)>
  <!ELEMENTFirstAired(PCDATA)>
  <!ELEMENTIMDB_ID(PCDATA)>
  <!ELEMENTzap2it_id(PCDATA)>
  <!ELEMENTid(PCDATA)>
]>
```

Donde los elementos importantes serían, por cada Serie:

- **seriesid:** identificador de la serie.
- **SeriesName:** nombre de la serie.
- **banner:** url del banner de la serie.
- **Overview:** descripción corta de la serie.

Updates

Se trata de una interfaz que permite el acceso las actualizaciones en la información de las series, episodios y banners que se han producido en un periodo que puede ser: diario, semanal, mensual o todas, para lo que se dispone de distintos ficheros que se generar dinámicamente para cada uno de los casos:

- Diario: http://thetvdb.com/data/updates/updates_day.xml
- Semanal: http://thetvdb.com/data/updates/updates_week.xml
- Mensual: http://thetvdb.com/data/updates/updates_month.xml
- Todas: http://thetvdb.com/data/updates/updates_all.xml

Esta interfaz se utiliza para actualizar diariamente (se utiliza updates_day.xml) la información de series y episodios.

El formato del XML sigue el siguiente DTD:

```
<!DOCTYPE Data [  
  <!ELEMENT Data (Series*, Episode*, Banner*)>  
  <!ELEMENT Series (id, time)>  
  <!ELEMENT Episode (id, Series, time)>  
  <!ELEMENT id (#PCDATA)>  
  <!ELEMENT time (#PCDATA)>  
  <!ELEMENT Series (#PCDATA)>  
>
```

Serie

Se trata de una interfaz que permite el acceso a la información de una serie específica en base a un identificador.

El formato de petición sería:

<http://thetvdb.com/api/121E6F2A75E6F1FC/series/<serieid>/all/es.xml>

Donde se debería sustituir <serieid> por el identificador de una serie concreta.

El formato del XML sigue el siguiente DTD:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE Data [
    <!ELEMENT Data (Series, Episode*)>
    <!ELEMENT Series (id, Genre, IMDB_ID, Language, Network, Overview, SeriesID, SeriesName,
Status, lastupdated, )>
    <!ELEMENT Episode (id, EpisodeNumber, SeasonNumber, lastupdated, FirstAired)>
    <!ELEMENT id (#PCDATA)>
    <!ELEMENT Genre (#PCDATA)>
    <!ELEMENT IMDB_ID (#PCDATA)>
    <!ELEMENT Language (#PCDATA)>
    <!ELEMENT Network (#PCDATA)>
    <!ELEMENT Overview (#PCDATA)>
    <!ELEMENT SeriesID (#PCDATA)>
    <!ELEMENT SeriesName (#PCDATA)>
    <!ELEMENT Status (#PCDATA)>
    <!ELEMENT lastupdated (#PCDATA)>
    <!ELEMENT EpisodeNumber (#PCDATA)>
    <!ELEMENT SeasonNumber (#PCDATA)>
    <!ELEMENT FirstAired (#PCDATA)>
]>
```

Episode

Se trata de una interfaz que permite el acceso a la información de un episodio específico en base a un identificador.

El formato de petición sería:

<http://thetvdb.com/api/121E6F2A75E6F1FC/episodes/<episodeid>/all/es.xml>

Donde se debería sustituir <episodeid> por el identificador de una serie concreta.

El formato del XML sigue el siguiente DTD:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE Data [
    <!ELEMENT Data (Episode)>
    <!ELEMENT Episode (id, EpisodeNumber, SeasonNumber, lastupdated, FirstAired)>
    <!ELEMENT id (#PCDATA)>
    <!ELEMENT EpisodeNumber (#PCDATA)>
    <!ELEMENT SeasonNumber (#PCDATA)>
    <!ELEMENT lastupdated (#PCDATA)>
    <!ELEMENT FirstAired (#PCDATA)>
]>
```

Banner

Se trata de una interfaz que permite el acceso a una banner específico, para poder ser cargado en una aplicación.

El formato de petición sería:

http://thetvdb.com/banners/_cache/<banner>

Donde se debería sustituir <banner> por lo que sea obtenido en una petición al servicio de GetSeries.

Propia

Las interfaces ofrecidas para los clientes/aplicaciones son:

- Obtención de nuevas emisiones.
- Inserción de nuevas series.
- Obtención de series ordenadas por calificación.
- Votación de un episodio.
- Obtener calificación de episodios y series.

Obtención de nuevas emisiones

Se trata de una interfaz que permite el acceso a series y episodios que han sido emitidos desde una fecha específica.

Se encarga de devolver aquellas series o episodios cuya fecha de emisión sea posterior a la pasada como última consultada pero anterior a la fecha actual.

El formato de petición sería:

<http://www.seriesnotifier.com/series/updates?type=<type>&time=<time>&series=<series>&format=<format>>

Donde:

- <type> se sustituiría por el tipo de actualizaciones pedidas, pudiendo ser series o episodes, siendo por defecto episodes. Parámetro optativo.
- <time> se sustituiría por el valor en timestamp de la última actualización solicitada. Parámetro obligatorio.
- <series> se sustituiría por una lista separada por comas de identificadores de series del que se quieran obtener actualizaciones. Parámetro obligatorio.
- <format> se sustituiría por el formato (xml o json) de la respuesta esperada. Es un parámetro opcional, el valor por defecto es json.

El formato del XML devuelto para el type episodes es la siguiente:

```
<!DOCTYPE Data [  
  <!ELEMENT updates (episodes*)>  
  <!ELEMENT episodes (id, seriename, serieid, seasonnumber, episodenumbr, date, rate,  
votes)>  
  <!ELEMENT id (#PCDATA) >  
  <!ELEMENT seriename (#PCDATA) >  
  <!ELEMENT serieid (#PCDATA) >  
  <!ELEMENT seasonnumber (#PCDATA) >  
  <!ELEMENT episodenumbr (#PCDATA) >  
  <!ELEMENT date (#PCDATA) >  
  <!ELEMENT rate (#PCDATA) >  
  <!ELEMENT votes (#PCDATA) >  
]>
```

Inserción de nuevas series

Se trata de una interfaz que permite insertar una serie en la base de datos en caso de que no se encuentre ya.

En todo caso, se comprueba que el valor introducido exista en TheTVDB, con lo que no se podría insertar cualquier valor.

El formato de petición sería:

<http://www.seriesnotifier.com/series/insert?id=<id>&name=<name>>

Donde:

- <id> se sustituiría por el identificador de la serie en TheTVDB. Parámetro obligatorio.
- <name> se sustituiría por el nombre de la serie.

Dicha interfaz se usará desde las aplicaciones/clientes para que puedan avisar al servicio de las series elegidas por sus usuarios y poder mantener los datos de estas en la base de datos.

No Devuelve ningún fichero de ningún formato. En caso de no poder insertarla, devuelve un error.

Obtención de series ordenadas por calificación

Se trata de una interfaz que permite acceder a un listado de las series que se encuentran en el sistema con mejor calificación por parte de los usuarios de TheTVDB. Se puede establecer la cantidad de series que se quieren recibir.

El formato de la petición sería:

<http://www.seriesnotifier.com/series/seriesbyrate?blocksize=<blocksize>>

Donde:

- <blocksize> se sustituiría por el valor del tamaño del bloque del listado que se espera recibir como máximo, el valor por defecto es 10.

Dicha interfaz se usará en las aplicaciones/clientes para obtener el listado de favoritos.

El formato del XML devuelto es la siguiente:

```
<!DOCTYPE Data [  
  <!ELEMENT series (serie*)>  
  <!ELEMENT serie (id, seriename, rate, votes)>  
  <!ELEMENT id (#PCDATA) >  
  <!ELEMENT seriename (#PCDATA) >  
  <!ELEMENT rate (#PCDATA) >  
  <!ELEMENT votes (#PCDATA) >  
>
```

Votación de un episodio

Se trata de una interfaz que permite votar un episodio, dicho voto será sumado a la calificación total del episodio y se usará para obtener la calificación del episodio en cuestión.

El formato de la petición sería:

<http://www.seriesnotifier.com/series/voteepisode?id=<episodeid>&vote=<value>>

Donde:

- <episodeid> se sustituiría por el identificador del episodio en cuestión a votar.
- <value> se sustituiría por el voto (de 0 a 10) de episodio en cuestión.

Dicha interfaz se usará en las aplicaciones/clientes para votar los episodios marcados como vistos.

Obtener calificaciones

Se trata de una interfaz que obtener la calificación de un episodio o serie en base a su identificador.

El formato de la petición sería:

<http://www.seriesnotifier.com/series/getrate?id=<id>&format=<format>>

Donde:

- <id> sería el identificador de la serie o episodio en cuestión.
- <format> sería el formato a solicitar, pudiendo ser episode o serie.

Dicha interfaz se utilizaría por los clientes para mostrar información de interés para los usuarios, en el caso de las series, en el detalle, y en el caso de los episodios, en el listado de series pendientes de ver, para poder seleccionar las mejor votadas.

Diagrama de clases

Web

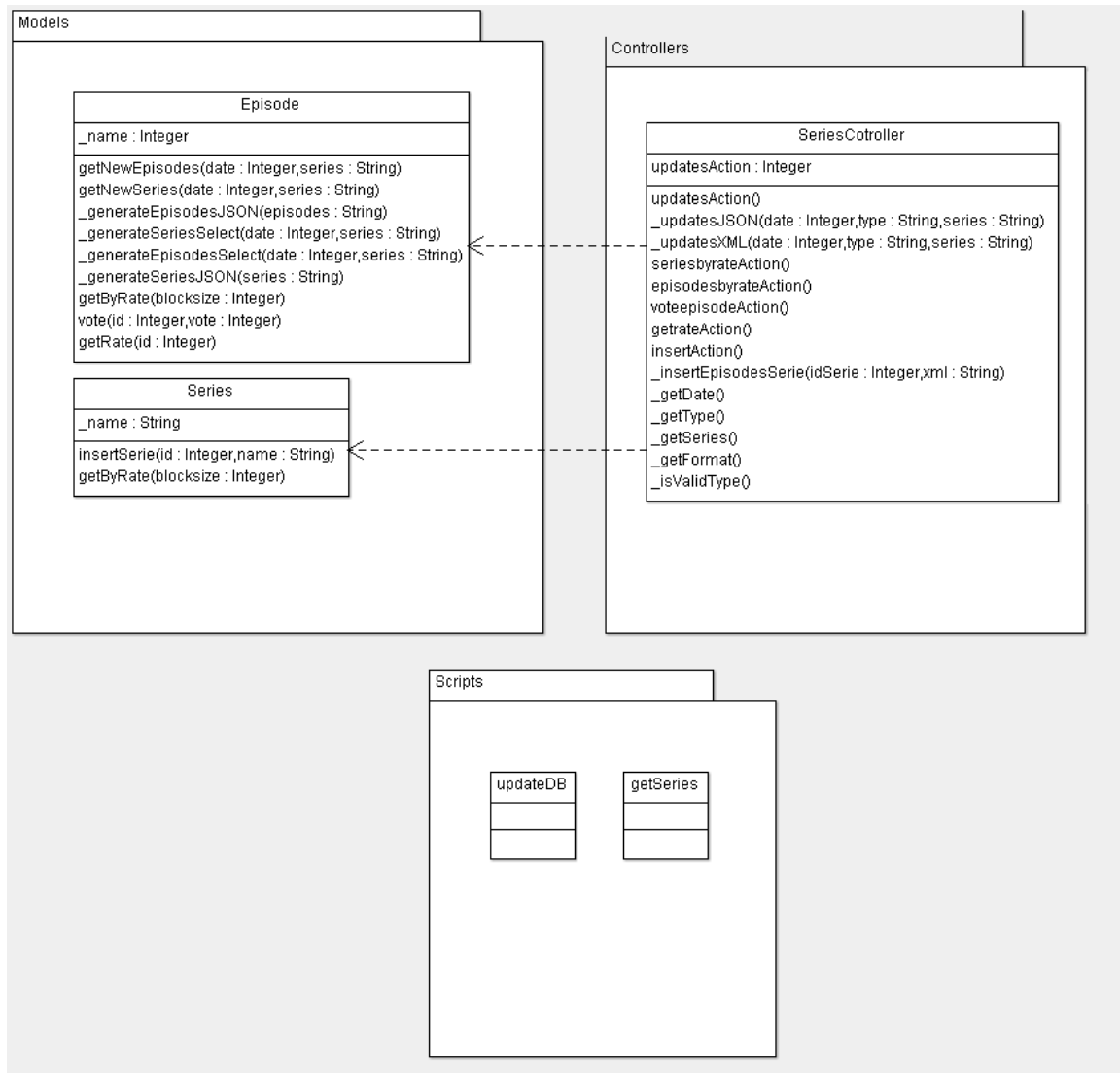


Ilustración 8: Diagrama de clases de la parte web

La parte web de la aplicación está dividida en tres componentes principales: Controlador, Modelos y Scripts.

El controlador se encarga de recibir las peticiones de acciones, redirigirlas al modelo adecuado y devolver el resultado en el formato esperado (XML o JSON).

Los modelos representan los objetos de la base de datos y se encargan de dar acceso a dichos objetos basándose en la lógica requerida, por lo que dispone de los métodos necesarios para las acciones disponibles en el controlador.

Los scripts se encargan de tareas puntuales o repetitivas que sirven para poblar la base de datos o actualizarla y por tanto hacer accesible los datos para su posterior acceso.



Página61

Vistas

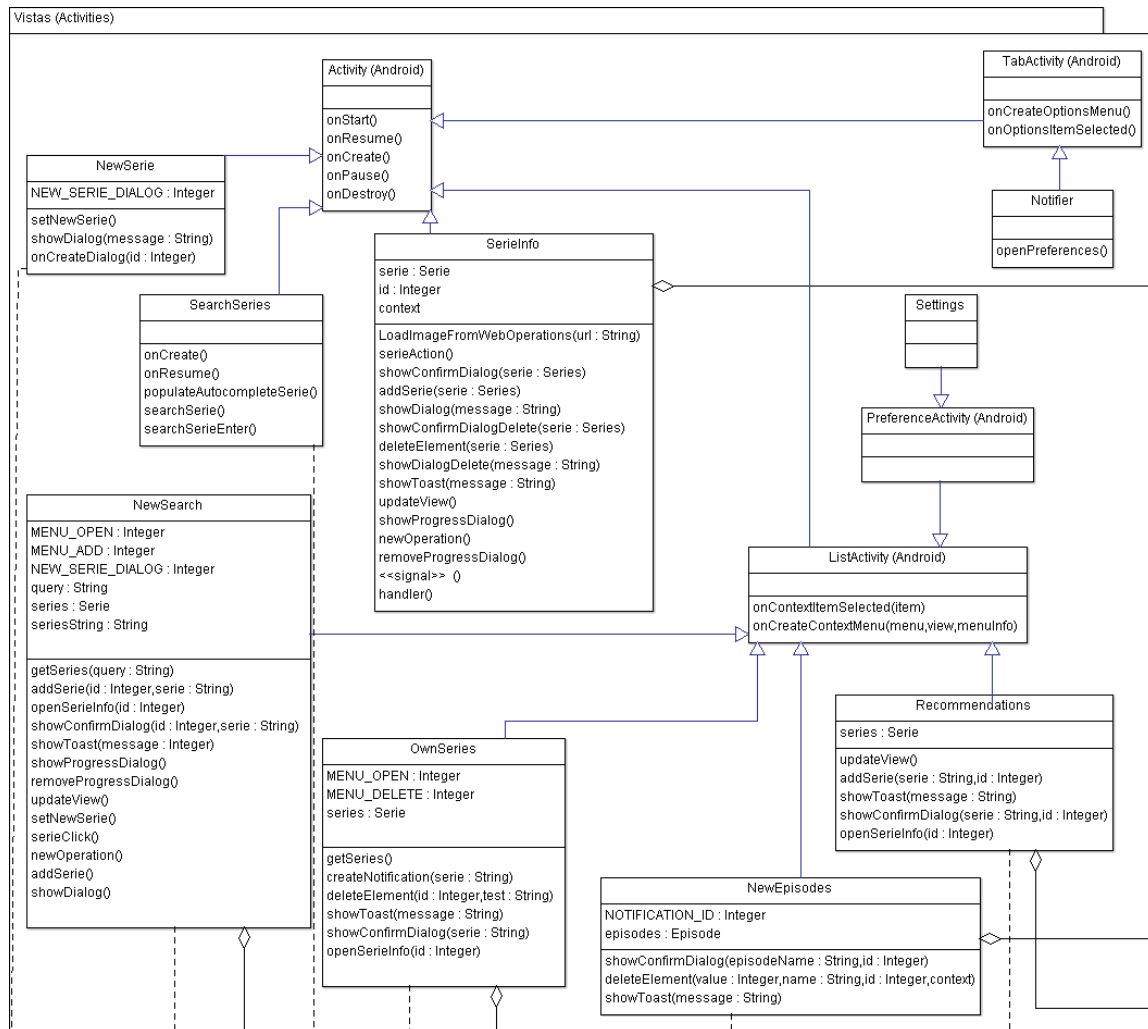


Ilustración 10: Diagrama de clases móvil: vistas

Las Activities en Android son la parte dinámica de las vistas y permiten definir qué tipos de objetos serán enviados o recibidos entre vistas, que comportamiento tendrá al abrir, cerrar o reanudar una “vista” así como gestionar el control de eventos dentro de los componentes de la vista, como por ejemplo qué hacer al pulsar un determinado botón o al presionar un elemento de una lista.

Todas las actividades en Android deben heredar de la superclase **Activity** o de alguna de sus subclases predefinidas: **TabActivity**, **ListActivity**, **PreferenceActivity**, ...

La aplicación se compone de la siguientes Actividades:

- **Notifier**: Actividad que hereda de **TabActivity**. Se trata de la actividad principal de la aplicación, que permite el acceso a otras actividades desde las pestañas que la componen: **SearchSeries**, **Recommendations**, **OwnSeries** y **NewEpisodes**. Permite el acceso a las preferencias de la aplicación desde cualquier **Activity** implementando el método `openPreferences()` de la clase **Activity**.

- **SearchSeries:** es la actividad de la pestaña activa por defecto y representa el formulario de búsqueda de series de la aplicación, permite introducir el nombre de una aplicación para buscarla. Hereda de Activity. Dispone de un método que se encarga de poblar los posibles resultados en el autocompletado (populateAutocompleteSerie) y del envío de la consulta de búsqueda (searchSerie) .
- **Recommendations:** se trata de una ListActivity en la que se muestra el listado de las series más populares para los usuarios. Sirviendo de posible recomendación al usuario. Dispone de métodos para controlar las acciones del usuario sobre el listado de series: añadir una serie a favorita, actualizar el listado, y para mostrar los mensajes de confirmación.
- **OwnSeries:** se trata de una ListActivity en la que se muestran las Series favoritas del usuario. Dispone de métodos para controlar las acciones del usuario sobre el listado de series: obtener la lista de series, eliminar una serie de favorita, actualizar el listado, y para mostrar los mensajes de confirmación.
- **NewEpisodes:** se trata de una ListActivity en la que se muestran los episodios que han sido emitidos de las series marcadas como favoritas, incluyendo sobre el episodio como cuándo fue emitido y que nota le han puesto otros usuario. Dispone de métodos para controlar las acciones del usuario sobre el listado de episodios: marcar un episodio como visto y para mostrar los mensajes de confirmación.
- **NewSearch:** se trata de una ListActivity que representa los resultados de una búsqueda, representando en formato lista aquellas series que concuerdan con la búsqueda realizar. Dispone de métodos para controlar las acciones del usuario sobre el listado de series: marcar una serie como favorita, navegar a la ficha de la serie, así como para mostrar los mensajes de confirmación.
- **SerieInfo:** se trata de una Activity que representa la ficha del detalle de una serie: nombre, detalle, imagen, estado. Dispone de métodos para controlar las acciones del usuario sobre la serie: marcar una como favorita o eliminar de favorita, así como control de eventos del usuario y mensajes de confirmación.
- **Settings:** se trata de una PreferenceActivity que representa las opciones de la aplicación, permitiendo gestionar al usuario cuando y con qué frecuencia quieren las notificaciones.

Modelos y Servicios

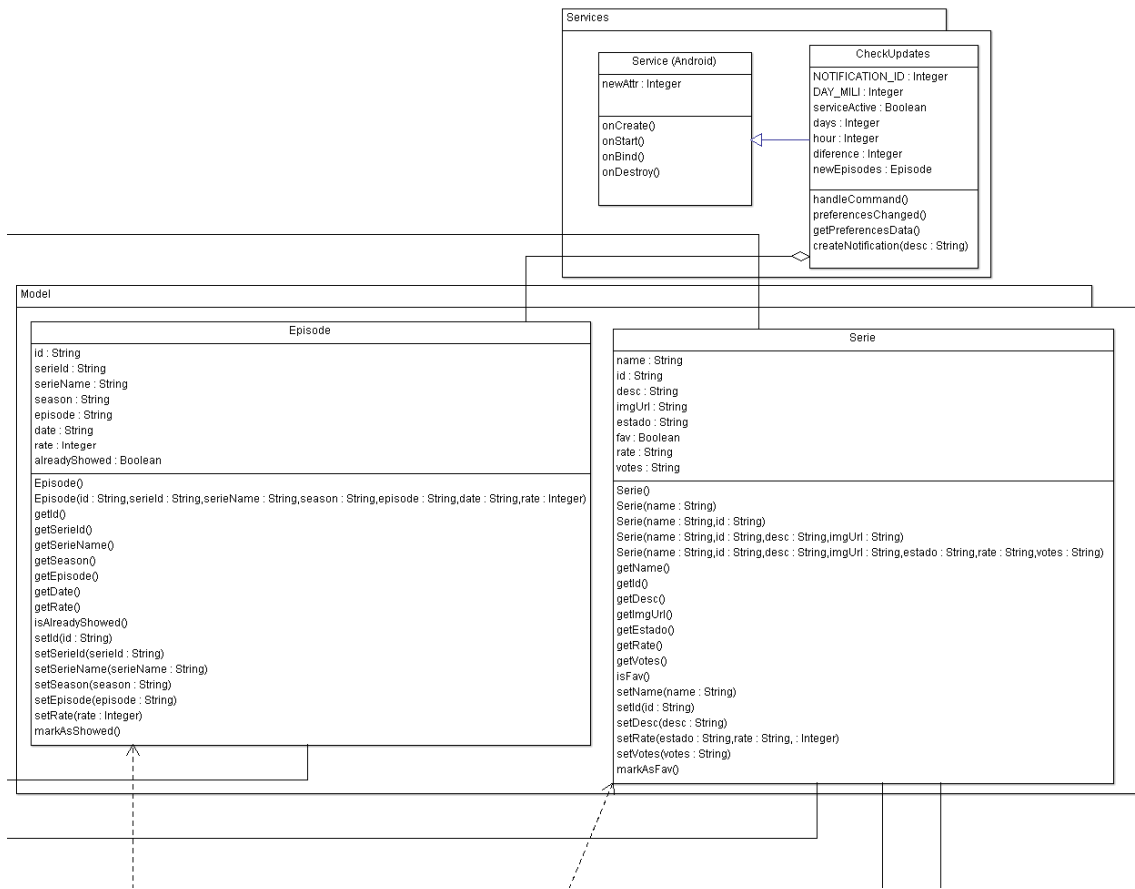


Ilustración 11:Diagrama de clases móvil: model y service

El único servicio, Service de Android, de la aplicación es una de las clases más importantes de la aplicación pues se encarga de realizar la comprobación periódica de existencia de nuevas series del catálogo de favoritas del usuario y realizar la notificación en la barra de notificaciones del sistema en caso de que exista alguna. Además debe gestionar los cambios en las preferencias de la aplicación, para poder cambiar la hora en la que se realizan las comprobaciones o su periodicidad.

En cuanto a los Modelos, representan los objetos de Series y Episodios, que son las entidades usadas en la aplicación. En ambos casos disponen de todos los atributos necesarios para poder representarse adecuadamente en las vistas (Activities) y los métodos de creación y acceso para todos los casos.

Utils

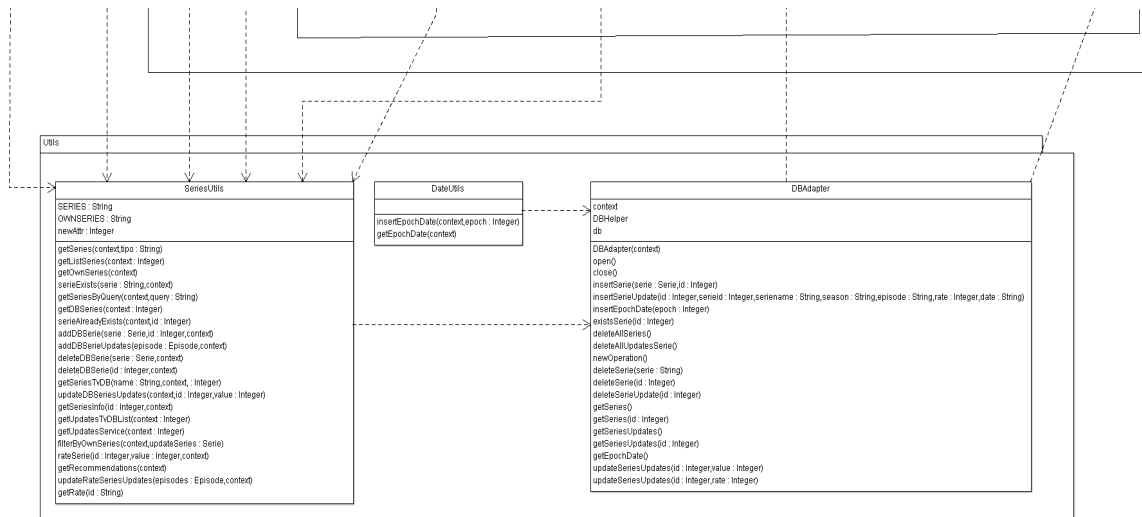


Ilustración 12: Diagrama de clases móvil: utils

Las clases de utilidades de la aplicación son contenedores de métodos de acceso y gestión de lógica compleja para limitar la complejidad de las Actividades y de los Modelos, de esta manera se gestionan los métodos de acceso a Servicios Web externos, los servicios web propios, a la base de datos, así como métodos para transformación de formatos y fechas.

La clase **DBAdapter** es un adaptador para la base de datos SQLite que dispone los dispositivos Android y que permite el almacenamiento de las series favoritas de los usuarios y de las actualizaciones (episodios pendientes de ver o ya marcados como vistos). Incluye métodos de creación del esquema, de borrado y creación de las tablas, y de obtención y borrado de datos en las 3 tablas de la Base de Datos.

La clase **SeriesUtils** reúne todos los métodos que acceden a datos sobre series y episodios, ya sean a través del Servicio Web Externo, del Servicio Web propio o para la base de datos, usando el **DBAdapter**.

La clase **DateUtils** reúne todos los métodos que acceden a datos sobre fecha de la aplicación: última actualización, así como conversión entre formatos.

Diseño de Interfaz

En este apartado se presentan los diferentes prototipos que se han realizado a lo largo del desarrollo del sistema.

Prototipo de bajo nivel

La primera versión del prototipo se realizó a mano, teniendo en mente los requisitos del sistema e ideando como podría ser el esquema básico de la aplicación y como se interactuaría con los diferentes componentes, se hicieron prototipos de unas cuantas pantallas solamente, pero se incluyen la mayor parte de las opciones e interacciones que compondrán la aplicación:

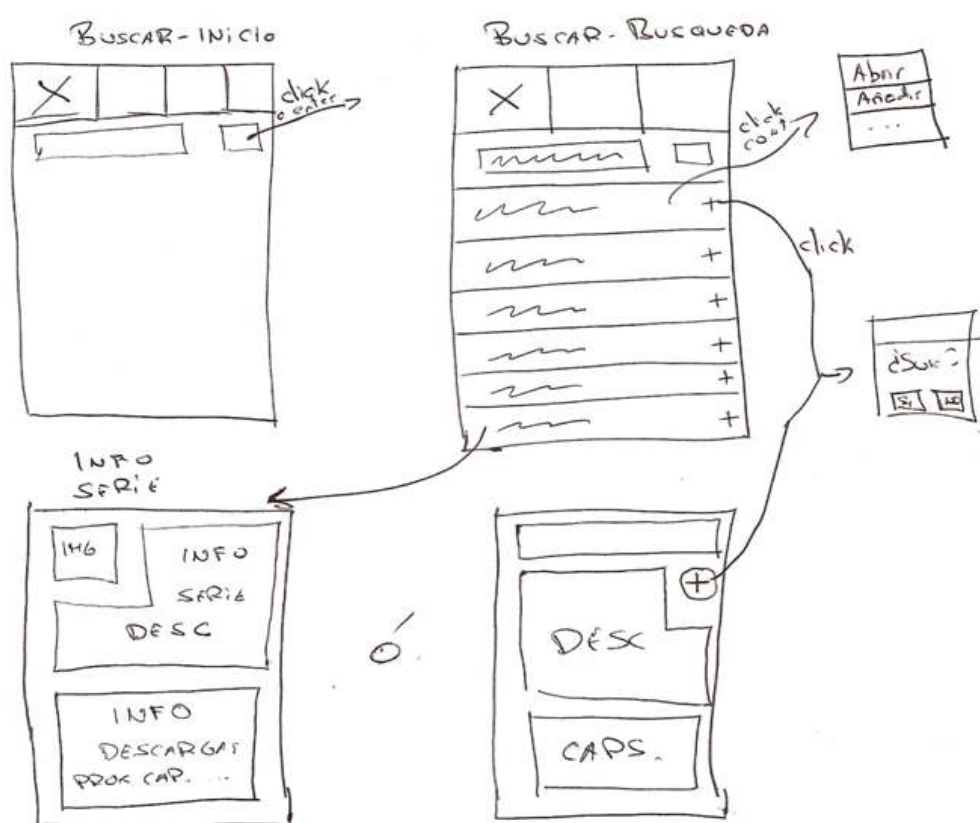


Ilustración 13: Prototipo de interacción, búsqueda

En la ilustración 15 se presenta el prototipo y el diseño de la interacción de la pantalla de búsqueda (Búsqueda - Inicio), del resultado de la búsqueda (Búsqueda - Búsqueda), las posibles interacciones dentro de la pantalla de resultados y un par de opciones para la ficha de una serie (Info Serie).

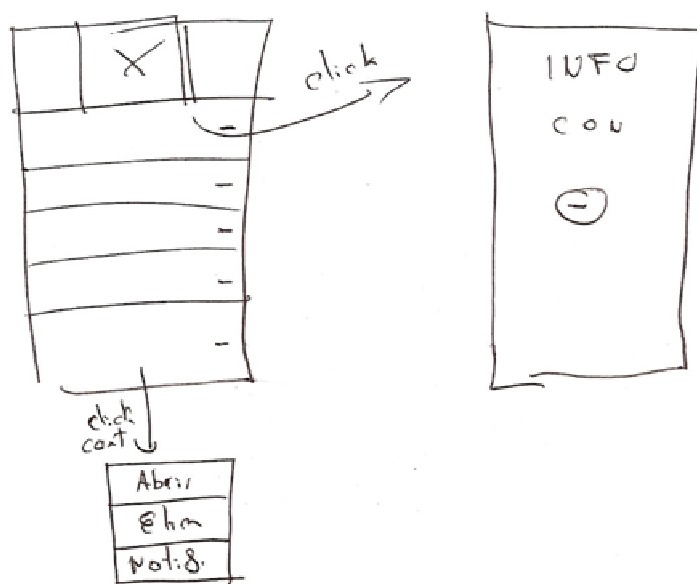


Ilustración 14: Prototipo de interacción, series favoritas

En la ilustración 16 se presenta el prototipo y el diseño de la interacción de la pantalla de Series Favoritas (Búsqueda - Inicio), las posibles interacciones dentro dicha pantalla, incluyendo la opción de ir a la ficha de la serie.

Prototipo de segundo nivel

La segunda versión del prototipo se realizó utilizando una herramienta de prototipado rápido denominada pencil, en esta segunda iteración se pudo hincapié en detallar los componentes que deberían ir en las principales pantallas de la aplicación, tal y como se detallan en los requisitos.

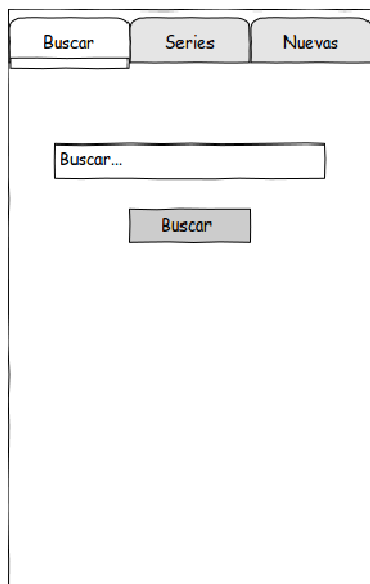


Ilustración 15: Pantalla principal

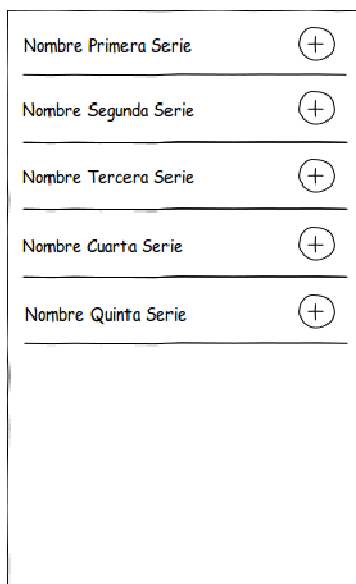


Ilustración 16: resultados búsqueda

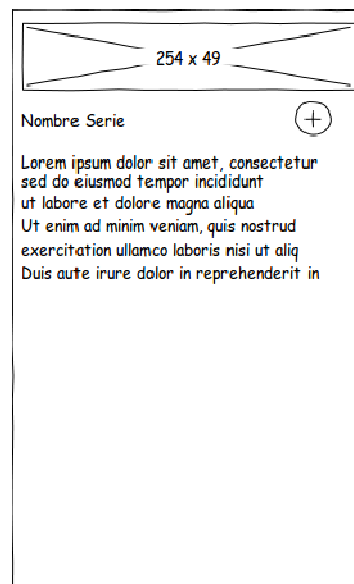


Ilustración 17: ficha serie

En las ilustraciones 17-19 se puede ver: la pantalla principal, que contiene el cuadro de búsqueda, una listado de series que representa el resultado de una búsqueda y la ficha de una series específica que se podría ver al pulsar sobre una de las series.

En el listado y la ficha se pueden ver en detalle cómo se deberán presentar los contenidos y que componentes tendrían cada pantalla: título y botón para añadir en el listado o imagen, título, botón y descripción en la ficha.

Buscar	Serie	Nuevas
Nombre Primera Serie		—
Nombre Segunda Serie		—
Nombre Tercera Serie		—
Nombre Cuarta Serie		—
Nombre Quinta Serie		—

Ilustración 18: Series Favoritas

Buscar	Serie	Nuevas
<input checked="" type="checkbox"/>	Nombre Primera Serie	SxEE
<input checked="" type="checkbox"/>	Nombre Segunda Serie	SxEE
<input checked="" type="checkbox"/>	Nombre Tercera Serie	SxEE
<input checked="" type="checkbox"/>	Nombre Cuarta Serie	SxEE
<input checked="" type="checkbox"/>	Nombre Quinta Serie	SxEE

Ilustración 19: Nuevos episodios

En las ilustraciones 20 y 21 se pueden ver los detalles de los listados de series favoritas del usuario y se nuevos episodios emitidos, así como los componentes que tendría cada listado: título y botón en caso de las favoritas y botón, título, número de temporada y capítulo en los nuevos episodios.

Prototipo de tercer nivel

Para la tercera versión del prototipo se hizo uso de la herramienta Pencil y de los componentes Android creados para ella, con lo que se consigue un prototipo cercano al resultado final. En esta versión se detallan tanto las pantallas como los mensajes de error, confirmación y espera de la aplicación:

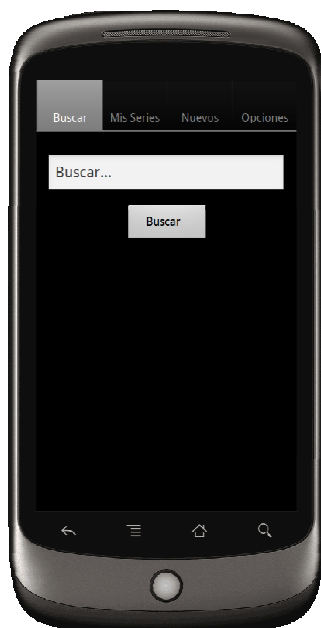


Ilustración 20: Pantalla principal



Ilustración 21: resultados de búsqueda



Ilustración 22: pantalla de progreso



Ilustración 23: ficha de serie

En las ilustraciones 22-25 se pueden ver las pantalla principal, cuadro de búsqueda, y las relacionadas con una búsqueda: resultados de búsqueda, pantalla de progreso en búsqueda y al cargar ficha y la ficha de una serie.

En ellas se puede ver en detalle como quedarán las pantallas en un dispositivo Android, además los botones de marcar como favorito se han sustituido por estrellas para mejorar la usabilidad de la aplicación

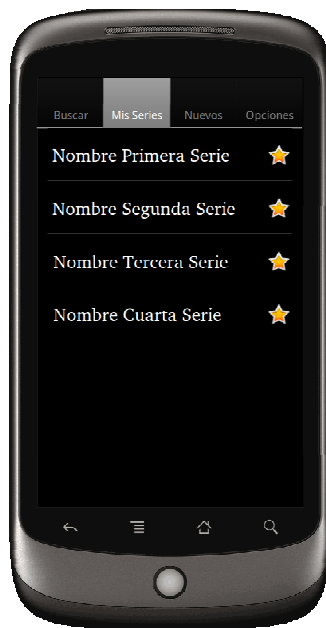


Ilustración 24: series favoritas



Ilustración 25: cuadro de confirmación

En las ilustraciones 26 y 27 se pueden ver en detalle la pantalla de series favoritas y el cuadro de dialogo de conformación, que valdrían para cualquier tipo de confirmación: confirmación de agregar a favorita, de eliminar de favorita

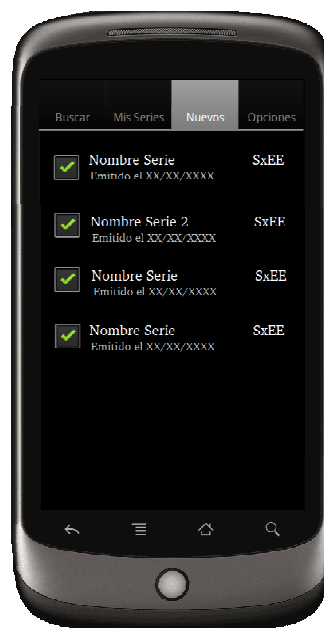


Ilustración 26: nuevos episodios

En la ilustración 28 se puede ver el detalle del listado de nuevos episodios emitidos, en el que se incluye un ticker para marcar como vistos, y la fecha de emisión con respecto al anterior prototipo.

Diseño final

La última iteración se trata del diseño real de la aplicación tal y como puede verse en un dispositivo Android o el emulador. Se han realizado capturas de todas las pantallas, mensajes de progreso y confirmación para detallar cada una de las pantallas.

Además de las pantallas anteriormente definidas, se han añadido:

- Top series: el listado de las series mejor valoradas por los usuarios que en los prototipos anteriores no se había añadido porque no se encontraba en la planificación su desarrollo.
- Página de resultado de búsqueda sin resultados
- Opciones: página con las opciones o preferencias de la aplicación.
- Mensajes de confirmación: mensaje de aplicación que indica al usuario que una acción ha sido realizada correctamente. Se mantiene durante un tiempo limitado a la vista.
- Diálogos de modificación de preferencias: en el que se puede seleccionar el valor para cada uno de los elementos modificables.



Ilustración 27: Pantalla principal, búsqueda



Ilustración 28: resultado de búsqueda



Ilustración 29: ficha de serie



Ilustración 30: mensaje de confirmación



Ilustración 31: series favoritas



Ilustración 32: top series



Ilustración 33: preferencias



Ilustración 34: dialogo preferencia, frecuencia

Implementación

En este apartado se detallan las diferentes herramientas utilizadas para la implementación de la aplicación, la organización del proyecto y los detalles sobre ciertas particularidades de la forma de implementación.

Eclipse[12]

Eclipse es un IDE de código abierto multiplataforma y multilenguaje que permite crear proyectos de software, permitiendo editar código, gestionar tareas, compartir y controlar versiones de los proyectos.

Permite la inclusión de plugins para extender su funcionalidad, con los que se puede añadir soporte a lenguajes de programación, sistemas de control de versiones, entornos de desarrollo móvil, gestión de pruebas o herramientas de integración continua.

En la realización del proyecto se ha usado la versión 3.6 denominada Helios, pero podría utilizarse cualquier versión posterior.

PDT Plugin[13]

PDT es un plugin para eclipse que permite el desarrollo de proyectos, gestionar el código y autocompletado de sintaxis para el lenguaje de programación PHP.

Este complemento ha sido utilizado para realizar la parte web del proyecto en el que se ha utilizado PHP como lenguaje de programación y Zend Framework como framework de desarrollo.

Subclipse Plugin[14]

Subclipse es un plugin para eclipse que permite integrar el sistema de control de versiones subversión dentro de eclipse para poder tener un control sobre las distintas versiones, herramienta muy útil para proyectos con equipos de varias personas o en el que se debe trabajar en varios equipos y entornos diferentes.

Este complemento ha sido utilizado para realizar el control de versiones tanto en la parte web como en la parte móvil (Android) del proyecto, utilizando repositorios independientes para cada proyecto.

Android SDK[15]

El SDK de Android se trata de un conjunto de herramientas para poder desarrollar, depurar y testear aplicaciones android desde la consola, así como instalar aplicaciones y gestionar los archivos en dispositivos físicos.

Este complemento es usado junto con el plugin para eclipse para realizar y probar la aplicación móvil desarrollada para dispositivos Android.

ADT Plugin[16]

ADT es un plugin para eclipse que permite integrar las herramientas de desarrollo (SDK) para dispositivos móviles Android, tal como la consola, la máquina virtual Dalvik y el emulador.

Este complemento ha sido utilizado para realizar y probar la aplicación móvil que ha sido realizada para dispositivos Android.

*Apache Server***[17]**

Apache es un servidor web de código abierto, multiplataforma y ampliamente extendido que permite la comunicación con múltiples clientes usando el protocolo HTTP.

Dicho servidor se usa para recibir y enrutar correctamente las peticiones de la parte web de la aplicación.

*Cron***[18]**

Cron es un administrador de procesos en segundo plano que ejecuta procesos en intervalos definidos de tiempo, la configuración de cuándo deben ejecutarse dichos procesos se gestiona a través de un fichero denominado crontab.

Cron se ha utilizado para ejecutar el proceso de actualización de información de series y episodios del API externo utilizado.

Organización del proyecto

La organización del proyecto se presenta de forma separada para cada uno de los componentes de la aplicación: la parte web y la parte móvil.

Web

El proyecto web, se realizó usando Zend Framework para facilitar ciertas tareas como el acceso a los datos de la base de datos y la separación de la lógica de negocio (Modelos) de el control de la peticiones (Controladores).

La estructura del proyecto estaría distribuida de la siguiente manera:

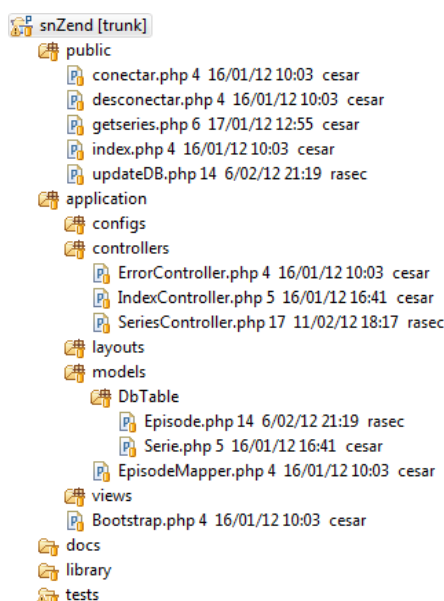


Ilustración 35: estructura del proyecto web

Donde se pueden distinguir los componentes normales de una aplicación Zend, y entre los que podríamos destacar los directorios:

- public: donde se encuentran los scripts PHP, entre ellos public.php que se encarga de inicializar la aplicación Zend y getseries.php y updateDB.php que se encargan de obtener la información del API externo.
- application: donde se encuentra la aplicación Zend:
 - controllers: donde se encuentran los controladores de la aplicación, principalmente SeriesController, que se encarga de gestionar las peticiones sobre series y episodios.
 - models: donde se encuentran las clases que gestionan el acceso a los datos de la base de datos y la lógica, destacan las clases Episode y Serie, que representan los dos tipos de datos básicos de la aplicación.

Android

El proyecto móvil, bastante más complejo que el web se realizó utilizando una estructura básica de proyectos Android según se genera utilizando el plugin para Eclipse:

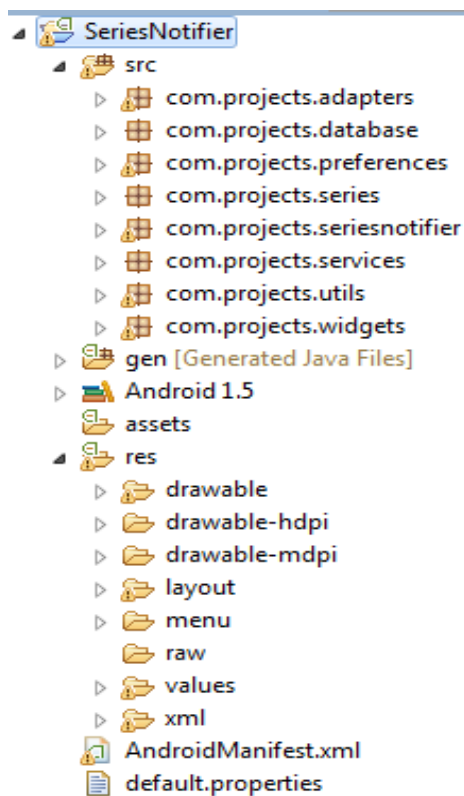


Ilustración 36: estructura de la aplicación android

Donde se puede distinguir los contenidos:

- src: directorio en el que se incluyen los archivos de código fuente Java: Activities, Services y cualquier clase Java, se puede ver en más detalle la estructura en:

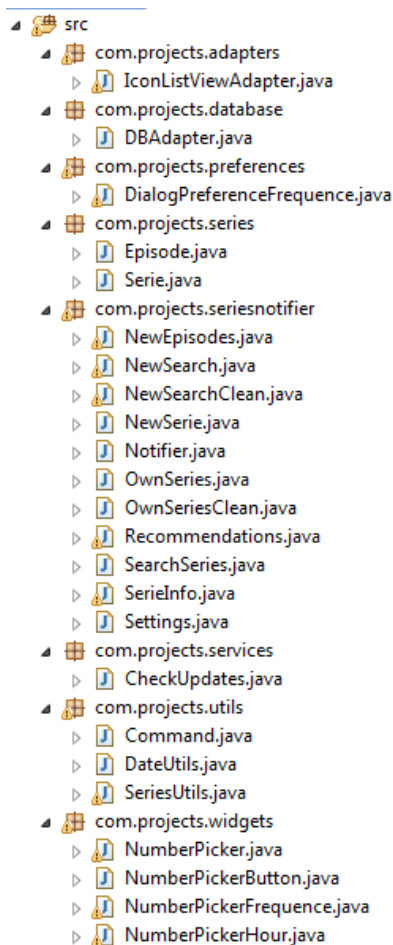


Ilustración 37: estructura de la aplicación android: directorio src

- gen: directorio que incluye los ficheros generados tras la compilación del proyecto, incluyendo el apk que puede usarse para instalar la aplicación.
- res: recursos del proyecto que no son código: imágenes (drawable), archivos xml de estructuras de Activities (layout), archivos con los textos de la aplicación (string.xml), de estilos (style.xml) y de configuración (settings.xml), se puede ver en más detalle la estructura en:

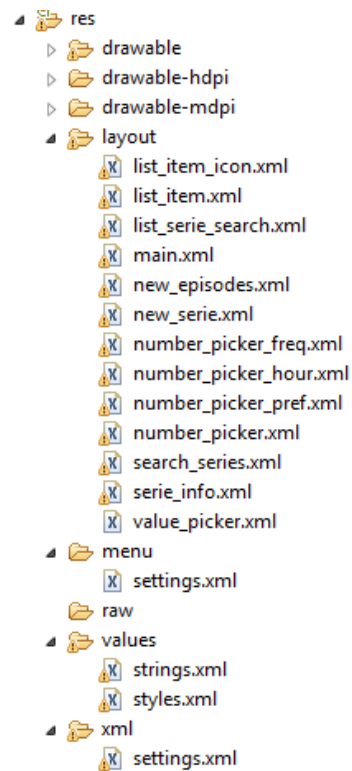


Ilustración 38: estructura de la aplicación android: directorio res

- AndroidManifest.xml: archivo manifiesto que incluye la información de la aplicación: nombre, valor de API mínimo, permisos de la aplicación, Activities y Servicios que va a tener.

5 Evaluación

En este apartado se presenta una evaluación de la solución elaborada, considerándose válida si se resuelven los problemas expuestos en el planteamiento del problema y si satisface los objetivos definidos.

Se ha optado por realizar una evaluación en dos fases, la primera es la realización de un plan de pruebas que asegure el cumplimiento de los casos de usos definidos y la segunda es la realización de una evaluación con usuario que valide la usabilidad de la aplicación.

5.1 Plan de Pruebas

Se ha diseñado un plan de pruebas que permiten verificar el cumplimiento de cada uno de los casos de uso previamente definidos y que por lo tanto permite verificar el correcto funcionamiento del sistema.

Cada prueba se compone de:

- Identificador: Número identificativo de la prueba
- Caso de uso asociado: identificador del caso de uso al que hace referencia.
- Descripción: información detallada del proceso necesario para realizar la prueba.

Identificador	Caso de Uso	Descripción
PR-001	CU-001	Se prueba a instalar la aplicación en un terminal compatible.
PR-002	CU-001	Se prueba a acceder a la aplicación a través del menú del terminal en el que se realizó la instalación. Debe abrirse la aplicación en la pantalla inicial (buscador)
PR-003	CU-002	Se verifica que se puede realizar la búsqueda de una serie usando una serie conocida. El resultado de la búsqueda debe ser un listado de series válidas
PR-004	CU-002	Se verifica que en caso de no introducir una búsqueda válida, la aplicación devuelve un mensaje de error.
PR-005	CU-003	Se verifica que al pulsar sobre el nombre de una serie en el listado de series obtenida como resultado de una búsqueda, se accede al detalle de dicha serie.
PR-006	CU-003	Se verifica que al pulsar sobre el nombre de una serie en el listado de series favoritas, se accede al detalle de dicha serie.
PR-007	CU-003	Se verifica que al pulsar sobre el nombre de una serie en el listado de top series, se accede al detalle de dicha serie.
PR-008	CU-004	Se verifica que al pulsar sobre el icono de la estrella de una de las series listadas en los resultados de búsqueda, si esta no se encuentra previamente marcada como favorita, se marca como favorita.

Identificador	Caso de Uso	Descripción
PR-009	CU-004	Se verifica que al pulsar sobre el icono de la estrella de una de las series listadas en el listado de series mejor valoradas (Top Series), si esta no se encuentra previamente marcada como favorita, se marca como favorita.
PR-010	CU-004	Se verifica que al pulsar sobre el icono de la estrella en el detalle de una serie, si esta no se encuentra previamente marcada como favorita, se marca como favorita.
PR-011	CU-004	Se verifica que siempre que se trata de marcar una serie como favorita, aparece un dialogo de confirmación.
PR-012	CU-004	Se verifica que siempre que se confirma el marcado de una serie como favorita, aparece un mensaje confirmando que la acción ha sido llevada a cabo correctamente.
PR-013	CU-005	Se verifica que al pulsar sobre el icono de la estrella de una de las series listadas en los resultados de búsqueda, si esta se encuentra previamente marcada como favorita, se elimina la serie de la lista de favoritas.
PR-014	CU-005	Se verifica que al pulsar sobre el icono de la estrella de una de las series listadas en la lista de series favoritas, se elimina la serie de la lista de favoritas.
PR-015	CU-005	Se verifica que al pulsar sobre el icono de la estrella en el detalle de una serie, si esta se encuentra previamente marcada como favorita, se elimina la serie de la lista de favoritas.
PR-016	CU-005	Se verifica que siempre que se trata de eliminar una serie de favoritas, aparece un dialogo de confirmación.
PR-017	CU-005	Se verifica que siempre que se confirma la eliminación de una serie de favoritas, aparece un mensaje confirmando que la eliminación ha sido llevada a cabo correctamente.
PR-018	CU-006	Se verifica que se puede acceder al listado de series mejor valoradas (Top Series) utilizando la pestaña específica de la aplicación.

Identificador	Caso de Uso	Descripción
PR-019	CU-007	Se verifica que se puede acceder al listado de mis series (series favoritas del usuario) utilizando la pestaña específica de la aplicación.
PR-020	CU-008	Se verifica que se puede acceder al listado episodios emitidos utilizando la pestaña específica de la aplicación.
PR-021	CU-008	Se verifica que se puede acceder al listado episodios emitidos a través del sistema de notificaciones de la aplicación.
PR-022	CU-009	Se verifica que se puede modificar la opción de frecuencia de comprobación del sistema de notificaciones a través de las preferencias de la aplicación.
PR-023	CU-010	Se verifica que se puede modificar la opción de hora de comprobación del sistema de notificaciones a través de las preferencias de la aplicación.
PR-024	CU-011	Se verifica que el sistema realiza notificaciones a la hora y con la frecuencia establecida en caso de encontrarse nuevos episodios emitidos.
PR-025	CU-011	Se verifica que se pueden ver las notificaciones de la aplicación sobre nuevos episodios emitidos en el menú de notificaciones del sistema.
PR-026	CU-012	Se verifica que se puede marcar un episodio como visto en el listado de episodios emitidos.
PR-027	CU-013	Se verifica que al marcar un episodio como visto se puede puntuar el episodio.
PR-028	CU-012	Se verifica que al marcar un episodio como visto se muestra un mensaje de confirmación.
PR-029	CU-014	Se verifica que se puede ver la nota media de la serie por parte de los usuarios en el detalle de dicha serie.
PR-030	CU-015	Se verifica que se puede ver la nota media de un episodio por parte de los usuarios de la aplicación en el listado de episodios emitidos.

Tabla 30: lista de casos de pruebas

A continuación se presentan los resultados de las pruebas de los casos de uso:

Prueba	001	002	003	004	005	006	007	008	009	010
Resultado	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito
Prueba	011	012	013	014	015	016	017	018	019	020
Resultado	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito
Prueba	021	022	023	024	025	026	027	028	029	030
Resultado	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito	Éxito

Tabla 31: resultados de las pruebas

En el que se puede comprobar que todas las pruebas resultaron satisfactorias y que por tanto se cumplen todos los casos de uso.

Por ello podemos concluir que la solución realizada cumple con las necesidades del usuario.

Evaluación con usuarios

La usabilidad[7] es un atributo de calidad que evalúa la facilidad de uso de las interfaces de usuario. La usabilidad es definida por cinco componentes:

- **Facilidad de aprendizaje:** facilidad con la que nuevos usuarios desarrollan una interacción efectiva con el sistema o producto.
- **Eficiencia:** Una vez que los usuarios han aprendido a usar el sistema, ¿Cuán rápido pueden realizar ciertas tareas?
- **Memorabilidad:** Cuando los usuarios vuelven a usar el sistema después de un tiempo sin haberlo usado, ¿Cómo de fácilmente pueden restablecer su nivel de manejo del sistema?
- **Errores:** ¿Cuántos errores cometen los usuarios, de qué nivel de severidad y cómo de fácilmente se recuperan de estos errores?
- **Satisfacción:** ¿Cómo de satisfactorio es usar el sistema?

Medir la usabilidad de una aplicación es un proceso complejo, puesto que la usabilidad no existe en un sentido absoluto sino que depende del contexto, de los usuarios objetivos y de las tareas que deben poder realizarse.

El objetivo de esta evaluación es medir la satisfacción de los usuarios al usar el sistema, por lo que se ha optado por usar la técnica '*cognitive walkthrough*'.

Dicha técnica consiste en:

1. Definir un conjunto de tareas clave a realizar usando la aplicación
2. Elegir un grupo de usuarios (potenciales usuarios objetivos de la aplicación) de diferentes niveles de conocimientos técnicos.
3. Poner a los usuarios a realizar las tareas.
4. Hacer a los usuarios que respondan a lo que se denomina un cuestionario *SUS* (*System Usability Scale*), que consisten en 10 preguntas con respuesta de escala 0-4 sobre usabilidad de la aplicación y que alterna las preguntas de tipo negativo y positivo para evitar las respuestas automáticas.

Para obtener la puntuación de cada cuestionario, se debe:

- Obtener sub-resultados:
 - Restar uno (1) a las preguntas impares.
 - Sumar cinco (5) y restar el valor de las preguntas pares.
- Sumar todos los sub-resultados.
- Multiplicar el valor de la suma por dos coma cinco (2.5).

El cuestionario a rellenar se puede encontrar en el anexo A.

Tareas

Las tareas que se han definido para realizar por parte de los usuarios son:

- Realizar la búsqueda de una serie y añadir dicha al listado de series del usuario.
- Recibir una notificación de la aplicación, leer la notificación y marcar como visto un episodio pendiente.

Participantes

Se ha seleccionado un grupo de 5 participantes, que reúnen las siguientes condiciones:

- Nivel de uso de Smartphone diferentes (alto, medio o bajo).
- Edad entre 20 y 35 años.
- Diferentes niveles de conocimientos técnicos (alto, medio, bajo o nulo).
- Experiencia en usabilidad (alta, media, baja o nula).

A continuación se muestra una tabla resumen con las características de los participantes:

	Uso Smartphones	Conocimientos técnicos	Experiencia usabilidad
Participante 1	Bajo	Nulo	Nula
Participante 2	Alto	Nulo	Nula
Participante 3	Medio	Medio	Alta
Participante 4	Alto	Alto	Baja
Participante 5	Bajo	Alto	Baja

Tabla 32: resumen participantes

Resultados

A continuación se muestra una tabla resumen con la puntuación obtenido en los cuestionarios, las respuestas por cada participante y el desglose de la puntuación se encuentra en el [Anexo V](#).

	Total	SUS Total
Participante 1	29	72.5
Participante 2	34	85
Participante 3	27	67.5
Participante 4	30	75
Participante 5	30	75

Tabla 33: resumen resultados

Conclusiones

Dado que los valores del SUS Total obtenido han sido, en su mayoría (4 de 5), superiores a 70, se puede concluir que la aplicación es usable y no requiere de un gran aprendizaje previo por parte de los usuarios. Incluso los usuarios con pocos conocimientos técnicos o con uso de smartphones limitado parecen sentirse cómodos con la aplicación según los resultados de los cuestionarios.

Como ideas de mejora y tras una consulta con algunos de los usuarios, se considera la introducción de ayuda dentro de la aplicación, ya sea en un apartado propio o usando un pequeño espacio dentro de cada sección.

6 Conclusiones

En este último apartado se presentan las conclusiones obtenidas a lo largo del desarrollo del proyecto. En él se analizan las aportaciones realizadas, las posibles líneas de mejora y ampliación, y las dificultades encontradas a lo largo del proyecto.

6.1 Aportaciones realizadas

La realización de este proyecto ha conllevado el desarrollo de una aplicación para dispositivos móviles inteligentes con sistema operativo Android, que permite a los usuarios gestionar notificaciones, buscar contenido, indicar sobre qué contenido quiere recibir notificaciones, configurar la frecuencia y hora de las notificaciones, recibir notificaciones sobre los nuevos contenidos y marcar qué contenido ha sido ya consultado o leído.

De esta manera se permite al usuario disponer de un sistema de notificaciones ampliamente configurable, que reduzca la distracción del usuario de sus tareas y que le permita estar al día de los cambios existentes.

Además se permite al usuario ser parte activa en el sistema, al permitir enviar su feedback en forma de nota.

6.2 Trabajos futuros

Como trabajo adicional para el futuro, se considera la opción de portar la aplicación a otros sistemas operativos móviles, como pudieran ser: iOS, Windows Phone, Blackberry, etc. Como la realización de una aplicación para cada dispositivo conllevaría mucho esfuerzo de aprendizaje y mucho tiempo, se contempla como mejor alternativa utilizar algunas de las herramientas que permite la creación de una aplicación y su distribución en diferentes Sistema Operativos móviles tales como PhoneGap o AppceleratorTitanium. En su defecto, podría crearse una aplicación web utilizando las tecnologías web emergentes incluidas dentro del estándar HTML5.

Además, para mejorar la experiencia del usuario, en base a las conclusiones obtenidas durante la evaluación de la aplicación, sería bueno añadir un sistema de recomendaciones de contenido en base a los gustos afines de otros usuarios de la aplicación.

Otra mejora en la experiencia del usuario obtenido durante la evaluación de la aplicación, sería la inclusión un sistema de ayuda o asistencia que permitiera al usuario conocer las posibilidades que ofrece la aplicación, así como la forma de acceder a dichas funcionalidades.

6.3 Problemas encontrados

A lo largo del desarrollo del proyecto han surgido diferentes problemas a los que ha habido que enfrentarse y darle una solución.

El primer problema surgió a la hora de decidir la versión mínima del sistema operativo Android que iba a ser soportada por aplicación. Esta decisión es relevante puesto que elegir una versión relativamente antigua limita las opciones o funcionalidades disponibles, pero una versión demasiado moderna reduce el público objetivo de la aplicación.

Otro de los puntos problemáticos fue conseguir la información necesaria que permitiera a la aplicación realizar las notificaciones, puesto que no existe casi ninguna API pública que permita el

acceso a dichos datos y finalmente hubo que realizar un sistema intermedio que solucionase ese problema.

6.4 Opiniones personales

La realización de este proyecto me ha llevado a conocer a fondo temas sobre los que me gustaría desarrollar mi carrera profesional, como es el desarrollo de aplicaciones en dispositivos móviles y otros dispositivos conectados, pero además se ha utilizado como temática uno de mis hobbies, las series de televisión, lo que ha facilitado mucho el que me involucrara más y mejor en el desarrollo del conjunto del proyecto, puesto que el primer interesado en la existencia de la aplicación soy yo mismo.

Ha sido un reto personal la conclusión de este proyecto, tanto por el tiempo dedicado, como por el hecho de que ha sido realizado compatibilizando un trabajo a tiempo completo, pero considero que ha sido positivo puesto que me ha abierto nuevas puertas dentro del mercado laboral y me ha hecho valorar y aprovechar más y mejor mi tiempo.

7 Bibliografía

Páginas o documentos electrónicos en la red

- [1] <http://lema.rae.es/drae/?val=notificacion>, accedido el 22 de Diciembre de 2012
- [2] <http://lema.rae.es/drae/?val=notificar>, accedido el 22 de Diciembre de 2012
- [4] http://en.wikipedia.org/wiki/Notification_system, accedido el 4 de Febrero de 2013.
- [6] http://es.wikipedia.org/wiki/Experiencia_de_usuario, accedido el 20 de Octubre de 2012.
- [8] <http://www.nngroup.com/about-user-experience-definition/>, accedido el 16 de Febrero de 2013.
- [9] <http://www.nngroup.com/articles/usability-101-introduction-to-usability/>, accedido el 20 de Noviembre de 2012.
- [10] http://es.wikipedia.org/wiki/Dise%C3%B1o_centrado_en_el_usuario
- [11] http://www.aipo.es/definicion_aipo.php
- [12] <http://www.nngroup.com/about/userexperience.html>, accedido en Noviembre 2012.
- [13] https://gs.statcounter.com/#mobile_os-ES-monthly-201101-201201, accedido en Febrero 2012.
- [14] <http://www.gartner.com/it/page.jsp?id=1848514>, accedido en Febrero 2012.
- [15] https://en.wikipedia.org/wiki/List_of_digital_distribution_platforms_for_mobile_devices#Operating_system-native_platforms, accedido en Febrero 2012.
- [16] <http://es.wikipedia.org/wiki/Android>, accedido en Febrero 2012.
- [17] [http://es.wikipedia.org/wiki/IOS_\(sistema_operativo\)](http://es.wikipedia.org/wiki/IOS_(sistema_operativo)), accedido en Febrero 2012.

- [18]http://es.wikipedia.org/wiki/Windows_Phone, accedido en Febrero 2012.
- [19]http://es.wikipedia.org/wiki/BlackBerry_OS, accedido en Febrero 2012.
- [20]<https://play.google.com/store/apps/details?id=com.googlecalendarfree.free2>, accedido en Febrero 2012.
- [21]<https://play.google.com/store/apps/details?id=com.google.android.gm>, accedido en Febrero 2012.
- [22]<https://play.google.com/store/apps/details?id=com.facebook.katana>, accedido en Febrero 2012.
- [23]<https://play.google.com/store/apps/details?id=com.handmark.tweetcaster>, accedido en Febrero 2012.
- [24]<http://eclipse.org>, accedido en Marzo 2012.
- [25]<http://www.eclipse.org/pdt/downloads/>, accedido en Marzo 2012.
- [26] <http://subclipse.tigris.org/>, accedido en Marzo 2012.
- [27]<http://developer.android.com/sdk/index.html>, accedido en Marzo 2012.
- [28]<http://developer.android.com/tools/sdk/eclipse-adt.html>, accedido en Marzo 2012.
- [29]<https://www.apache.org/>, accedido en Marzo 2012.
- [30] [http://es.wikipedia.org/wiki/Cron_\(Unix\)](http://es.wikipedia.org/wiki/Cron_(Unix)), accedido en Abril 2012.

Tesis Doctorales

- [3]Mohamed, N.,Romaithi, A., Samahi, E., Kendi, M. and Jabrie, E.: “Generating Web Monitors”, The College of Information Technology (United Arab Emirates University). 2008.
- [5]D. Scott McCrickard, C. M. Chewar, Jacob P. Somervell and Ali Ndiwalana: “A Model for Notification System Evaluation”, Virginia Polytechnic Institute and State University, November 2003
- [7]Arhippainen, L., Tähti, M.: “Empirical Evaluation of User Experience in Two Adaptive Mobile Application Prototypes”. University of Oulu, 2003

- [11] Kankainen, A. :“Thinking model and tools for understanding user experience related to information appliance product concept” Tesis Doctoral, Helsinki University of Technology, 2002.

Anexo I. Control de versiones

Este anexo contiene las distintas versiones por las que ha ido pasando el documento a lo largo de su realización:

Versión	Fecha	Páginas	Modificaciones
0.1	21/03/2012	13	Añadida estructura inicial. Añadido estado del arte
0.2	30/04/2012	45	Modificado estado del arte (0.1) Añadido el apartado de Gestión de proyecto Software y el presupuesto
0.3	22/05/2012	54	Modificada Gestión de proyecto Software. (0.2) Añadida Definición de requisitos de la aplicación. Añadido listado de casos de uso de la aplicación.
0.4	18/06/2012	96	Añadido Diseño de la aplicación. Añadido Diseño de la interfaz de la aplicación.
0.5	12/07/2012	93	Modificada estructura del apartado de diseño de la interfaz (0.4). Añadido apartado de implementación de la aplicación.
0.6	31/08/2012	87	Reorganización del apartado de solución de la aplicación (0.3) Modificada estructura del apartado de diseño de la interfaz (0.4) Cambios en el apartado de implementación (0.5)
0.7	23/09/2012	89	Añadido apartado de evaluación.
0.7.1	24/11/2012	94	Añadido apartado de evaluación con usuarios.
0.8	02/12/2012	98	Añadido apartado de introducción
0.8.1	19/01/2013	96	Modificado el apartado de introducción (0.8).
0.8.2	5/02/2013	96	Modificado el apartado de introducción (0.8.1).
0.8.3	17/02/2013	96	Modificado el apartado de introducción (0.8.2).
0.9	02/03/2013	99	Añadido apartado Resumen
0.9.1	17/03/2013	99	Modificado apartado Resumen (0.9) Añadido apartado conclusiones
0.9.2	02/04/2013	99	Modificado apartado Resumen (0.9.1) Modificado apartado Conclusiones (0.9.1)
0.9.3	08/04/2013	100	Modificaciones apartado Conclusiones (0.9.2)
1	12/04/2013	100	Versión para entregar a tribunal.

Tabla 34: control de versiones

Anexo II. Seguimiento de proyecto fin de carrera

Para llevar a cabo las distintas tareas de la forma más eficiente posible, es necesario realizar una planificación mediante la cual organizar el tiempo del cual disponemos.

Es importante remarcar que la realización de este trabajo se hace con una limitación importante de horarios dada la situación actual del autor, por lo que la planificación inicial, que es la que exponremos en este anexo ha sufrido variaciones considerables a lo largo de la realización del proyecto.

Forma de seguimiento

Para un correcto seguimiento del proyecto, se han ido realizando entregas periódicas del documento y de la aplicación a lo largo del desarrollo de la aplicación, para obtener una posterior validación de los tutores del proyecto.

Por otro lado, para aclaración de dudas, toma de requisitos, presentación de prototipos, etc... se han ido realizando reuniones de seguimiento con los tutores.

Planificación inicial

A continuación se muestra una tabla con la planificación inicial realizada sobre las distintas fases del proyecto

Tarea	Sub-Tarea	Estimación
Estudio de alternativas	Estudio de plataformas	3 días
	Estudio de aplicaciones	3 días
	Total	6 días
Análisis	Definición de requisitos	5 días
	Especificación de requisitos	10 días
	Total	15 días
Diseño	Diseño del modelo de datos	3 días
	Diseño de la capa de servicios	7 días
	Diseño de la lógica de negocio	7 días
	Diseño de la interfaz de usuario	8 días
	Total	25 días
Documentación	Documentación fase inicial	7 días
	Documentación de implementación y pruebas	8 días
	Total	15 días
Implementación	Implementación de capa de servicios	10 días
	Implementación de lógica de negocio	15 días
	Implementación de interfaz de usuario	15 días
	Total	40 días
Pruebas	Pruebas unitarias	5 días
	Pruebas funcionales	5 días
	Pruebas de usabilidad	5 días
	Total	15 días
TOTAL		116 días

Ilustración 39: planificación inicial

Planificación final

A continuación se muestra una tabla con la planificación final real de las distintas fases del proyecto.

Tarea	Sub-Tarea	Estimación
Estudio de alternativas	Estudio de plataformas	6 días
	Estudio de aplicaciones	4 días
	Total	10 días
Análisis	Definición de requisitos	10 días
	Especificación de requisitos	16 días
	Total	26 días
Diseño	Diseño del modelo de datos	3 días
	Diseño de la capa de servicios	10 días
	Diseño de la lógica de negocio	8 días
	Diseño de la interfaz de usuario	10 días
	Total	31 días
Documentación	Documentación fase inicial	15 días
	Documentación de implementación y pruebas	17 días
	Total	31 días
Implementación	Implementación de capa de servicios	11 días
	Implementación de lógica de negocio	14 días
	Implementación de interfaz de usuario	17 días
	Total	42 días
Pruebas	Pruebas funcionales	3 días
	Pruebas de usabilidad	8 días
	Total	11 días
TOTAL		151 días

Ilustración 40: Planificación final

Como se aprecia existe una desviación de todos los tiempos con respecto a la planificación inicial expuesta en el apartado anterior de este anexo, siendo más significativa en el apartado de documentación.

Comparativa

En la siguiente grafica se presenta la variación en tiempos que han sufrido cada una de las fases de desarrollo de la aplicación, encontrándose los valores en días.

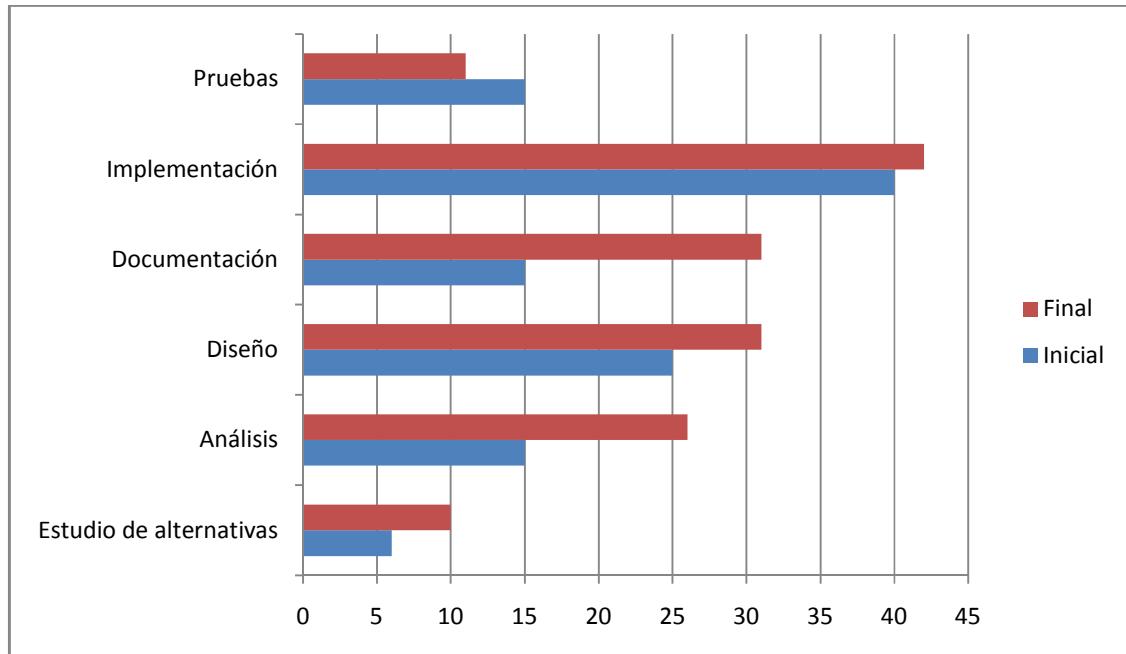


Ilustración 41: Desviación de la realización del proyecto

Anexo V. Cuestionarios de evaluación

Participante 1

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
2. I found the system unnecessarily complex	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
1 2 3 4 5					
3. I thought the system was easy to use	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
1 2 3 4 5					
5. I found the various functions in this system were well integrated	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
6. I thought there was too much inconsistency in this system	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	1	2	3	4	5
8. I found the system very cumbersome to use	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
1 2 3 4 5					
9. I felt very confident using the system	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5

Puntuación:

1ª 0 2ª 4 3ª 4 4ª 3 5ª 2 6ª 2 7ª 4 8ª 4 9ª 2 10ª 4

Total: 29

SUS Total: 29*2.5= 72.5

Participante2

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
2. I found the system unnecessarily complex	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
1 2 3 4 5					
3. I thought the system was easy to use	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
1 2 3 4 5					
5. I found the various functions in this system were well integrated	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
6. I thought there was too much inconsistency in this system	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
8. I found the system very cumbersome to use	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
1 2 3 4 5					
9. I felt very confident using the system	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	1	2	3	4	5

Puntuación:

1ª 3 2ª 3 3ª 3 4ª 4 5ª 3 6ª 4 7ª 3 8ª 4 9ª 3 10ª 4

Total: 34

SUS Total: 34*2.5= 85

Participante3

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block; text-align: center;">✓</div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>
	1	2	3	4	5
2. I found the system unnecessarily complex	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block; text-align: center;">✓</div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>
1 2 3 4 5					
3. I thought the system was easy to use	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block; text-align: center;">✓</div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>
	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block; text-align: center;">✓</div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>
1 2 3 4 5					
5. I found the various functions in this system were well integrated	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block; text-align: center;">✓</div>
	1	2	3	4	5
6. I thought there was too much inconsistency in this system	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block; text-align: center;">✓</div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>
	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block; text-align: center;">✓</div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>
	1	2	3	4	5
8. I found the system very cumbersome to use	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block; text-align: center;">✓</div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>
1 2 3 4 5					
9. I felt very confident using the system	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block; text-align: center;">✓</div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>
	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block; text-align: center;">✓</div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>	<div style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></div>
	1	2	3	4	5

Score:

1^a 2 2^a 3 3^a 2 4^a 3 5^a 4 6^a 3 7^a 2 8^a 3 9^a 2 10^a 3

Total: 27

SUS Total: 27*2.5= 67.5

Participante4

	Strongly disagree		Strongly agree		
1. I think that I would like to use this system frequently	1	2	3	4	5
	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2. I found the system unnecessarily complex	1	2	3	4	5
	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3. I thought the system was easy to use	1	2	3	4	5
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5. I found the various functions in this system were well integrated	1	2	3	4	5
	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6. I thought there was too much inconsistency in this system	1	2	3	4	5
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
8. I found the system very cumbersome to use	1	2	3	4	5
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9. I felt very confident using the system	1	2	3	4	5
	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5
	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Puntuación:

1ª 2 2ª 4 3ª 3 4ª 4 5ª 2 6ª 3 7ª 3 8ª 3 9ª 2 10ª 4

Total: 27

SUS Total: 30*2.5= 75

Participante5

	Strongly disagree			Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4
	1	2	3	4
2. I found the system unnecessarily complex	1	2	3	4
	1	2	3	4
3. I thought the system was easy to use	1	2	3	4
	1	2	3	4
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4
	1	2	3	4
5. I found the various functions in this system were well integrated	1	2	3	4
	1	2	3	4
6. I thought there was too much inconsistency in this system	1	2	3	4
	1	2	3	4
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4
	1	2	3	4
8. I found the system very cumbersome to use	1	2	3	4
	1	2	3	4
9. I felt very confident using the system	1	2	3	4
	1	2	3	4
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4
	1	2	3	4

Puntuación:

1ª 3 2ª 3 3ª 3 4ª 4 5ª 4 6ª 3 7ª 2 8ª 3 9ª 2 10ª 3

Total: 27

SUS Total: 30*2.5= 75